

Percursos de maquinagem 2D: mesa CNC e robô industrial

Luís Carlos Santos Oliveira

Dissertação do MIEM

Orientador: Manuel Romano dos Santos Pinto Barbosa

Coorientador: Paulo Augusto Ferreira de Abreu



Mestrado Integrado em Engenharia Mecânica

Especialização de Automação

Junho de 2018

À minha família

Resumo

A globalização é responsável por diversas modificações no mundo, destacando-se o avanço tecnológico acelerado e consequentemente uma intensa robotização na produção industrial. No caso particular da maquinagem, a utilização de robôs industriais é também uma área em crescimento. A sua grande flexibilidade e relação custo/benefício torna-os uma solução complementar a máquinas CNC, que têm um espaço de trabalho limitado e custo elevado. No entanto, são ainda muitas as barreiras a ultrapassar para que os robôs assumam um papel preponderante na execução de operações de maquinagem, dado que apresentam baixa exatidão e rigidez e, ainda, limitação de *software* específico.

Ao longo deste trabalho é realizada uma análise comparativa de parâmetros e funções de programação, disponíveis em dois sistemas distintos: mesa XY comandada por CNC e robô industrial. Um dos objetivos principais é identificar as implicações da utilização destas funções na execução de percursos de maquinagem, do tipo fresagem e respetivas velocidades de avanço. Igualmente pretende-se também avaliar a facilidade e eficiência da programação em cada sistema. A mesa XY é comandada por um CNC Fagor® 8055M e o robô industrial é um robô ABB® IRB2400, usando apenas como ferramenta um dispositivo para traçar os percursos (caneta) numa folha de papel. Para além das funcionalidades de programação *online* de cada sistema, usaram-se ainda as aplicações de *software* para programação e simulação offline: Mastercam®, RobotMaster® e RobotStudio®. Recorrendo a este *software* foram programados movimentos elementares, lineares e circulares, de modo a tirar ilações para posteriormente aplicar em movimentos mais complexos, associados a estratégias de maquinagem para abertura de caixas (*pocket*). Os parâmetros escolhidos para comparação tiveram em consideração o interesse nas operações de maquinagem, em se manter uma velocidade de avanço constante ao longo do percurso de maquinagem. No robô, o principal parâmetro influenciador da trajetória e do perfil da velocidade, é o parâmetro *zone*, que permite condicionar o desvio de posição relativamente à posição programada com uma menor variação de velocidade. No que concerne ao CNC foram usadas as funções arredondamento de aresta (G05) e *look-ahead* (G51), revelando estas ser fortemente dependentes da velocidade programada. Finalmente, foram implementados e testados os programas para executar percursos associados a oito tipos de estratégias de maquinagem disponíveis no Mastercam®. A análise concentrou-se nas capacidades de pós-processamento dos dois sistemas, relativamente a tempos de execução, extensão dos programas, distâncias de maquinagem e percursos realizados.

2D machining paths: CNC XY-table and industrial robot

Abstract

Globalization is responsible for several modifications in the world, accelerating technological advances and consequently an intense robotization in industrial production. In the machining area the use of industrial robots is a growing technology. Due to their great flexibility and cost/benefit relation they are an alternative to CNC machines, which have a limited working space and high cost. However, there are many barriers to overcome for robots to assume a preponderant role, given that they have low accuracy and rigidity, and limitation of specific *software*.

Throughout this work a comparative analysis of programming parameters and functions is performed in two systems: XY table commanded by CNC and industrial robot. One of the main objectives is to identify the implications of using these functions in the execution of milling-type machining paths and their feed rates. It is also intended to evaluate the ease and efficiency of programming in each system. The XY table is controlled by a CNC Fagor® 8055M and a industrial robot is an ABB® IRB2400, using only a tool to trace the path (pen) on a sheet of paper. In addition to the online programming features of each system, software applications for offline programming and simulation were also used: Mastercam®, RobotMaster® and RobotStudio®. Using this software, elementary, linear and circular movements were programmed, so as to draw lessons to later apply in more complex movements, associated to strategies of machining pockets. The parameters chosen for comparison took into account the interest in the machining operations in maintaining a constant feed rate along the machining path. In the robot, the main influencing parameter of the trajectory and the speed profile is the *zone* parameter, which allows to condition the position deviation from the programmed position with a lower speed variation. As for the CNC, the roundness of edge (G05) and look-ahead (G51) were used, revealing that these are strongly dependent on the programmed speed. Finally, the programs were implemented and tested to execute paths associated with eight types of machining strategies available in Mastercam®. The analysis focused on the post-processing capabilities of the two systems in terms of path execution times, machining part program lengths, machining distances and machining paths.

Agradecimentos

Gostaria de agradecer ao meu orientador e coorientador, o Professor Manuel Romano Barbosa e o Professor Paulo Abreu, respetivamente, por toda a paciência, atenção e conhecimentos científicos transmitidos a este projeto, sendo que tiveram como prioridade a indicação do caminho mais acertado.

À minha família, por me ter dado condições, apoio e carinho incondicionais ao longo de toda a minha vida.

Por fim gostaria de reconhecer o contributo, para este trabalho, de todos os meus amigos e colegas que desde o início me acompanharam. Sem eles, todo este percurso não seria possível.

Índice

1	Introdução	1
1.1	Objetivos	2
1.2	Estrutura da dissertação	2
2	Programação e maquinagem CNC.....	5
2.1	Sistemas de comando numérico.....	5
2.2	Principais características dos CNC atuais.....	7
2.3	Sistemas CAM	14
3	Maquinagem com robôs industriais	19
3.1	Aplicações de robôs industriais	19
3.2	Programação de robôs.....	27
4	Características dos sistemas utilizados	35
4.1	Elementos principais da mesa XY e controlador CNC.....	35
4.2	Funcionalidades de configuração e programação do CNC 8055M	40
4.3	Funcionalidades de programação dos movimentos do robô IRB 2400	46
4.4	Síntese	49
5	Casos de estudo e testes realizados.....	51
5.1	Procedimento e metodologia.....	51
5.2	Caso de estudo 1: Análise do movimento linear com 3 pontos	56
5.3	Caso de estudo 2: Análise do movimento em U	68
5.4	Caso de estudo 3: Análise do movimento circular.....	75
5.5	Caso de estudo 4: Análise de estratégias de maquinagem	78
6	Conclusões e perspectivas de trabalhos futuros	81
	Referências	83
	ANEXO A: Funções de programação do CNC	86

Índice de Figuras

Figura 2.1 - Interface do Fanuc 30i, 31i e 32i (à esquerda) e 0i (à direita) [5].	9
Figura 2.2 - Várias soluções da Siemens no setor do controlo numérico (adaptado de [7]).	11
Figura 2.3 - Módulos do Mastercam 2018 (adaptado de [12]).	15
Figura 2.4 - Módulos do NX.	16
Figura 3.1 - Configuração típica de um robô [14].	19
Figura 3.2 - Distribuição dos robôs vendidos em 2016 por setor industrial [15].	20
Figura 3.3 - Simulação no <i>software</i> SprutCAM (à esquerda) e operação de acabamento (à direita) [25].	25
Figura 3.4 - Solução robótica de polimento de pás de turbinas a gás [28].	26
Figura 3.5 - Processo de criação de esculturas [30].	27
Figura 3.6 - Programação por guiamento (à esquerda) e programação via consola (à direita) [32].	28
Figura 3.7 - Ambiente de simulação do Robotmaster [34].	31
Figura 3.8 - Funcionalidades avançadas do menu <i>Optimization</i> [35].	32
Figura 4.1 - Sistema Fagor 8055M.	35
Figura 4.2 - Driver de velocidade com feedback do motor [36].	36
Figura 4.3 - Eixo linear do movimento X (servomotor, fuso e guias de apoio).	37
Figura 4.4 - Função <i>Oscilloscope</i> disponível no CNC.	39
Figura 4.5 - Teste ballbar realizado para uma velocidade de 1200 mm/min (à esquerda) e o círculo obtido (à direita).	40
Figura 4.6 - Trajetória circular tangente à trajetória (G08) [40].	42
Figura 4.7 - Trajetória circular definida mediante três pontos (G09).	43
Figura 4.8 - Interpolação helicoidal.	43
Figura 4.9 - Função de aresta viva (G07) [40].	44
Figura 4.10 - Arredondamento de aresta controlado (G50) [40].	44
Figura 4.11 - Função de arredondamento de aresta G05 [40].	45
Figura 4.12 - Efeito da função <i>look-ahead</i> na velocidade de avanço (adaptado de [2]).	45
Figura 4.13 - Elemento terminal do robô.	46
Figura 4.14 - Evidência do parâmetro zone na descrição de trajetórias (adaptado [41]).	48
Figura 4.15 - Instrução <i>AccSet</i> [41].	48
Figura 4.16 - Célula robótica virtual utilizada nos casos de estudo.	49
Figura 4.17 - Perfil de velocidades do TCP para as várias parametrizações de <i>AccSet</i> .	49
Figura 5.1 - Análise dos casos de estudo do robô.	52
Figura 5.2 - Análise dos casos de estudo do CNC.	53
Figura 5.3 - Percurso implementado - linha.	56

Figura 5.4 – Representação parcial dos 3 percursos em linha realizados pelo robô para 20 mm/s (<i>fine</i> , <i>z1</i> e <i>z10</i>).....	57
Figura 5.5 – Representação parcial dos 3 percursos em linha realizados pelo robô para 200 mm/s (<i>fine</i> , <i>z1</i> e <i>z10</i>).....	57
Figura 5.6 – Excerto dos perfis de velocidade do TCP durante a execução de uma linha para 20 mm/s (<i>z1</i> e <i>z10</i>).....	58
Figura 5.7 – Excerto dos perfis de velocidade do TCP durante a execução de uma linha para 200 mm/s (<i>z1</i> e <i>z10</i>).....	58
Figura 5.8 – Representação parcial dos 4 percursos obtidos usando as funções G01, G05 e G51 (E1.0 e E10.0) para 20 mm/s.	58
Figura 5.9 – Representação parcial dos 4 percursos obtidos usando as funções G01, G05 e G51 (E1.0 e E10.0) para 200 mm/s.	59
Figura 5.10 – Excerto do perfil da velocidade de avanço na execução de um percurso em linha para 20 mm/s usando a função arredondamento de aresta (G05).....	59
Figura 5.11 – Excerto do perfil da velocidade de avanço na execução de um percurso em linha para 200 mm/s usando a função arredondamento de aresta (G05).....	59
Figura 5.12 – Excerto dos perfis da velocidade de avanço na execução de percursos em linha para 20 mm/s e usando a função <i>look-ahead</i> (G51).	60
Figura 5.13 – Excerto dos perfis da velocidade de avanço na execução de percursos em linha para 20 mm/s e usando a função <i>look-ahead</i> (G51).	60
Figura 5.14 – Percurso implementado – canto.	61
Figura 5.15 – Representação parcial das posições do TCP para oito percursos (<i>fine</i> , <i>z1</i> , <i>z5</i> e <i>z10</i>).	62
Figura 5.16 – Excerto dos perfis de velocidade do TCP durante a execução de um percurso em canto para 20 mm/s (<i>z1</i> e <i>z10</i>).	62
Figura 5.17 – Excerto dos perfis de velocidade do TCP durante a execução de um percurso em canto para 200 mm/s (<i>z1</i> e <i>z10</i>).	63
Figura 5.18 – Representação parcial dos 4 percursos obtidos recorrendo ao uso das funções G01 e G05 (20 e 200 mm/s).	64
Figura 5.19 – Desvio aproximado da posição em função da velocidade de avanço, para o percurso em canto.....	64
Figura 5.20 – Excerto do perfil da velocidade de avanço na execução de um percurso em canto para 20 mm/s, usando a função de arredondamento de aresta (G05).	65
Figura 5.21 - Excerto do perfil da velocidade de avanço na execução de um percurso em canto para 200 mm/s, usando a função de arredondamento de aresta (G05).	65
Figura 5.22 – Representação parcial dos percursos obtidos para 20 mm/s recorrendo à função <i>look-ahead</i> (G51).....	66
Figura 5.23 – Representação parcial dos percursos obtidos para 200 mm/s recorrendo à função <i>look-ahead</i> (G51).....	66
Figura 5.24 – Excerto dos perfis da velocidade de avanço na execução de percursos em canto para 20 mm/s, usando a função <i>look-ahead</i> (G51).....	67
Figura 5.25 – Excerto dos perfis da velocidade de avanço na execução de percursos em canto para 200 mm/s, usando a função <i>look-ahead</i> (G51).....	67

Figura 5.26 – Percurso em U de base reta (à esquerda) de base circular (à direita).....	69
Figura 5.27 – Representação parcial dos percursos em U de base reta para 20 mm/s (<i>fine</i> e <i>z1</i> a <i>z100</i>).	70
Figura 5.28 – Representação parcial dos percursos em U de base reta para 200 mm/s (<i>fine</i> e <i>z1</i> a <i>z100</i>).	70
Figura 5.29 – Representação parcial dos percursos em U de base reta para 20 mm/s (G01, G05 e G51).	70
Figura 5.30 – Representação parcial dos percursos em U de base reta para 200 mm/s (G01, G05 e G51).	71
Figura 5.31 – Percurso e respetivo programa, implementados no RI – movimento em U de base circular.	72
Figura 5.32 – Percurso e respetivo programa, implementados no CNC – movimento em U de base circular.	72
Figura 5.33 – Representação parcial dos percursos em U de base circular para 20 mm/s (<i>fine</i> e <i>z1</i> a <i>z100</i>).	73
Figura 5.34 – Representação parcial dos percursos em U de base circular para 200 mm/s (<i>fine</i> e <i>z1</i> a <i>z100</i>).	73
Figura 5.35 – Representação parcial dos percursos em U de base circular para 20 mm/s (G01/G02, G05 e G51).	74
Figura 5.36 – Representação parcial dos percursos em U de base circular para 200 mm/s (G01/G02, G05 e G51).	74
Figura 5.37 – Percurso programado para o caso do robô.	75
Figura 5.38 – Percursos circulares para 20 mm/s (<i>fine</i> e <i>z1</i> a <i>z100</i>).	76
Figura 5.39 – Percursos circulares para 200 mm/s (<i>fine</i> e <i>z1</i> a <i>z100</i>).	76
Figura 5.40 – Percursos circulares para 200 mm/s (<i>fine</i> e <i>z1</i> a <i>z100</i>) com um parâmetro <i>AccSet 10,10</i>	77
Figura 5.41 – Percursos circulares para 20 mm/s (G02, G05 e G51).	77
Figura 5.42 – Percursos circulares para 200 mm/s (G02, G05 e G51).	77
Figura 5.43 - Trajetórias obtidas no robô (em cima) e CNC (em baixo).	80

Índice de Tabelas

Tabela 2.1 - Vantagens e desvantagens de um sistema CNC relativamente a máquinas convencionais (adaptado de [3]).....	7
Tabela 2.2 - Funções CNC disponíveis nos vários modelos de controlo numérico do fabricante Fanuc (adaptado de [6]).....	10
Tabela 2.3 - Número e tipo de eixos controlados pelos vários modelos SINUMERIK [8]	12
Tabela 2.4 - Funções CNC disponíveis nos vários modelos de controlo numérico SINUMERIK (adaptado de [9])	13
Tabela 2.5 - Estratégias de maquinagem disponibilizadas pelo Mastercam 2018	16
Tabela 3.1 - Exemplos de aplicações de maquinagem usando robôs [16]	21
Tabela 3.2 - Robôs vs máquinas CNC nas aplicações de maquinagem (adaptado de [18]).....	22
Tabela 3.3 - Exemplos de robôs para operações de maquinagem	24
Tabela 3.4 - <i>Software</i> proprietário de programação e simulação <i>offline</i> de robôs	28
Tabela 3.5 - <i>Software</i> de programação genérico de robôs.....	29
Tabela 3.6 - <i>Software</i> CAM com pós-processador integrado.....	29
Tabela 3.7 - <i>Software</i> de pós-processamento	30
Tabela 4.1 - Exemplos de variáveis que podem ser atribuídas a um dos quatro canais do <i>Oscilloscope</i>	39
Tabela 4.2 - Modo de execução de trajetórias no sistema CNC e no robô.....	50
Tabela 5.1 - Casos de estudo analisados	54
Tabela 5.2 - Parâmetros específicos para cada caso de estudo.....	55
Tabela 5.3 - Tempos de execução do percurso em linha.....	56
Tabela 5.4 – Tempos de execução do percurso em canto	61
Tabela 5.5 – Tempos de execução do percurso em U de base reta	69
Tabela 5.6 – Tempos de execução do percurso em U de base circular	73
Tabela 5.7 – Tempos de execução de realização de uma circunferência	75
Tabela 5.8 - Comparação de estratégias de maquinagem CNC/RI	79

Siglas

CAD	Computer Aided Design
CAE	Computer Aided Engineering
CAM	Computer Aided Manufacturing
CNC	Computer Numeric Control
DDS	Digital Drive System
MMI	Man Machine Interface
NCK	Numerical Control Kernel
SERCOS	Serial Real-Time Communication System
TCP	Tool Center Point
RI	Robô Industrial

1 Introdução

Diante de um mundo cada vez mais globalizado, torna-se fundamental para um fabricante obter vantagem competitiva face aos seus rivais. Com o crescente aumento da competitividade entre mercados e o crescimento sustentável, a inovação e a elevada produtividade industrial passaram a ser condições *sine qua non* para o bom desempenho de qualquer empresa. O tempo de desenvolvimento de um produto deve ser reduzido, sendo que, escolher um processo de fabrico adequado, de entre os disponíveis, pode-se tornar um processo complexo tendo em conta as múltiplas alternativas e variáveis em jogo.

A realização de protótipos consiste na produção de um número reduzido de unidades (às vezes uma) de produto, pelo que se justifica recorrer a máquinas com grande capacidade de adaptação para permitir executar produtos diferentes, como por exemplo, máquinas ferramenta de comando numérico (CNC). No caso particular de operações de maquinagem, a flexibilidade de robôs industriais, as suas semelhanças com máquinas CNC nos componentes que integram e a geração de trajetórias que permitem realizar, torna-os uma alternativa interessante, devido ao seu baixo custo e à disponibilidade de ferramentas de programação de robôs para operações de maquinagem. Para além da sua utilização nos processos de fabrico subtrativos (*e. g.* maquinagem por arranque de apara), a capacidade dos robôs industriais na execução de trajetórias pode ainda ser explorada para implementação de processos de fabrico aditivos, também conhecidos por impressão 3D. A flexibilidade da utilização de robôs industriais, neste tipo de aplicações, tem ainda interesse num contexto cada vez mais alargado de flexibilidade dos sistemas de produção, relativamente à customização de produtos.

As máquinas CNC, estando otimizadas para tarefas de maquinagem, contam com características e funções que foram criadas de raiz tendo como principal foco a maquinagem. Opostamente, o robô foi criado com um propósito mais abrangente em termos de aplicações e, como tal, a programação e estrutura foram gradualmente adaptadas para esta aplicação. Os robôs industriais possuem uma estrutura física menos rígida e consequentemente têm menor exatidão na realização de operações de maquinagem. Contudo, a crescente tendência para a utilização dos robôs em aplicações nesta área, quer tirando partido do desenvolvimento e adaptação dos sistemas de CAD/CAM existentes, quer pelo desenvolvimento por parte dos

fabricantes de RI de estruturas físicas e *software* de programação e simulação, estão a permitir ultrapassar os problemas atuais.

Neste contexto torna-se especialmente importante analisar e comparar as funcionalidades de ambos os sistemas (CNC e robô industrial) na execução de operações de maquinagem.

1.1 Objetivos

Tendo em conta os equipamentos e sistemas disponíveis, este trabalho tem como principal finalidade a comparação e a avaliação das implicações de diferentes parametrizações na definição de estratégias de maquinagem em 2D, permitidas pela tecnologia CNC e pela maquinagem com robôs industriais.

Assim, a realização deste trabalho compreenderá:

- A análise e identificação das funções específicas de programação para maquinagem com sistemas CNC;
- A identificação e caracterização das funcionalidades de maquinagem do sistema CNC existente;
- A análise e configuração do *software Mastercam e Robotmaster*;
- A definição das estratégias de maquinagem e a variação dos respetivos parâmetros;
- A simulação, implementação, teste e consequente análise de resultados obtidos, com os percursos de maquinagem definidos para as geometrias correspondentes aos casos de estudo.

1.2 Estrutura da dissertação

Os temas abordados e o trabalho que foi preconizado, traduziram-se numa estrutura de dissertação que culminou em seis capítulos, incluindo o atual, referente à introdução.

O capítulo 2, intitulado, *Programação e maquinagem CNC*, faz uma caracterização abrangente dos sistemas de comando numérico, incluindo os seus principais elementos e implicações da sua utilização para os processos de maquinagem. Paralelamente, é dada ênfase às funcionalidades específicas de programação e configuração permitidas pela tecnologia CNC mais atual e pelos sistemas CAM.

O capítulo 3, *Maquinagem com robôs industriais*, apresenta uma caracterização de robôs industriais (RI), em geral, no que diz respeito aos seus componentes principais e campo de aplicação na realização de operações de maquinagem. Estas incluem não só operações tradicionalmente realizadas por RI (*e. g.* rebarbagem, corte, *etc.*), como também operações de maquinagem clássicas das máquinas ferramenta (*e. g.* fresagem). Um aspeto fulcral nestas aplicações relaciona-se com a análise e disponibilidade de ferramentas de suporte à simulação e elaboração dos programas de maquinagem, a partir de sistemas CAD/CAM.

Relativamente ao capítulo 4, *Características dos sistemas utilizados*, são enumeradas as particularidades da mesa de dois eixos (XY) e do respetivo controlador (CNC), bem como da célula robótica. Nesta caracterização incluem-se os sistemas de suporte à elaboração e simulação/teste dos respetivos programas de maquinagem, as possibilidades destes sistemas implementarem os testes idealizados e de recolherem a informação necessária, com vista à sua avaliação.

No tocante ao capítulo 5 intitulado, *Casos de estudo e testes realizados*, apresentam-se os percursos de maquinagem a realizar, os elementos geométricos que são usados para a sua definição, a combinação de parâmetros de maquinagem e funções de programação. São ainda apresentados e comparados os resultados obtidos para as medidas de eficiência estabelecidas.

Finalmente no que concerne ao capítulo 6, *Conclusões e perspetivas de trabalhos futuros*, apresentam-se as principais ilações que resultaram da análise global deste trabalho e algumas sugestões para trabalhos futuros.

2 Programação e maquinagem CNC

Os processos de maquinagem convencionais estão na base do aparecimento dos sistemas de CN e do seu posterior desenvolvimento. Neste capítulo apresentam-se as vantagens da tecnologia CNC para os processos de maquinagem, as características principais dos sistemas mais evoluídos e ainda as alternativas para elaboração dos programas de CN, dando especial ênfase aos sistemas CAD/CAM.

2.1 Sistemas de comando numérico

O objetivo principal da tecnologia CNC é permitir sincronizar os eixos de movimento principais de uma máquina ferramenta (*e. g.* fresadora), de modo a executar peças com formas de geometria complexa. Atualmente a sua aplicação e funcionalidades estendem-se a vários processos de fabrico nomeadamente [1]:

- Maquinagem convencional por arranque de apara (*e. g.* fresagem, torneamento e retificação);
- Maquinagem não convencional (*e. g.* eletroerosão, corte por jato de água e eletroquímicos);
- Trabalhos em chapa (*e. g.* quinagem, puncionamento e corte por laser).

Em geral podem considerar-se processos que envolvem a necessidade de controlar a posição de uma peça, ou material a trabalhar, em relação a uma ferramenta ou ao ponto de aplicação da energia característica do processo de fabrico. As variáveis a controlar pelo CNC consistem, assim, na posição e velocidade dos eixos de movimento (*feed axis*) e parâmetros de aplicação da fonte de energia principal do processo de fabrico (*e. g.* *spindle*).

No caso particular da fresagem, os parâmetros de maquinagem consistem na velocidade de avanço (*feed rate*), na penetração ou profundidade de corte e na velocidade de corte, relacionada com a velocidade de rotação da ferramenta (*spindle speed*) [2].

A complexidade das formas obtidas depende do número e tipo de eixos de movimento, com possibilidade de interpolação simultânea, sendo a mais simples a maquinagem com dois eixos (2D), mas também é igualmente comum, a maquinagem com três (3D), quatro (4D) e até cinco eixos (5D). A tecnologia CNC permite ainda o controlo simultâneo de um número ainda superior de eixos de movimento, permitindo desenvolver máquinas ferramenta capazes de executar simultaneamente várias operações de maquinagem na mesma peça, ou em peças diferentes, conjugando grande flexibilidade e elevada produtividade.

Em relação ao grau de automação na execução do processo de maquinagem, a tecnologia CNC teve implicações a dois níveis. Por um lado, no controlo e automatização de funções e movimentos auxiliares, desde a mudança automática de ferramenta, da peça a trabalhar, sistemas de lubrificação e refrigeração, sistemas de segurança e até monitorização do estado da máquina. Por outro lado, no controlo das variáveis do processo de fabrico com ajuste automático de parâmetros de corte, tais como, nas velocidades de avanço ou de corte, em função do material ou secção de apara a cortar.

A grande liberdade que a tecnologia CNC permite, em relação às formas possíveis de serem obtidas por maquinagem, está condicionada pelos meios disponíveis para elaboração dos programas de maquinagem (CNC). Desde a programação no controlador da própria máquina (*online*), usando códigos normalizados ou funcionalidades de programação gráfica, até à programação em *software* específico (gráfico) de programação CNC, fora da máquina (CAD e CAM), diferentes níveis de suporte ou facilidade na elaboração e teste dos programas CN são possíveis, como será abordado nas secções seguintes. Neste contexto, existem dois métodos diferenciados de programação de máquinas CNC:

- Programação *online* ou diretamente na máquina – *Manual Data Input* (MDI) - O programador, baseado na geometria da peça a fabricar, define o percurso da ferramenta e as condições de maquinagem da máquina. É usado em casos simples de programação ou eventuais modificações de programas;
- Programação *offline* - O programador importa o desenho da peça para o *software*. Posteriormente, é gerado o percurso da ferramenta através das ferramentas CAM. Não existe necessidade de calcular a trajetória da ferramenta, atribuindo esta tarefa ao computador. Este método é muito utilizado, atualmente em particular quando as peças envolvem maquinagem de superfícies complexas.

Analisando a Tabela 2.1, pode-se inferir que as vantagens do sistema CNC superam as desvantagens, daí a sua grande aplicabilidade e versatilidade em várias áreas da indústria, em detrimento das técnicas tradicionais.

Tabela 2.1 - Vantagens e desvantagens de um sistema CNC relativamente a máquinas convencionais (adaptado de [3])

<i>Vantagens</i>	<i>Desvantagens</i>
Menor tempo de execução	Grande investimento inicial
Eliminação dos erros do operador	Maior custo de manutenção
Vida mais longa da ferramenta	Operários mais especializados
Ideal para lotes mais pequenos	
Flexibilidade na alteração do desenho	
Melhor previsão de custos	

Do ponto de vista funcional, um sistema CNC é constituído por [2]:

- Interface homem-máquina (MMI – *Man Machine Interface*) - permite mostrar o estado da máquina e oferecer funções para editar, modificar ou elaborar o programa de maquinagem;
- CNC (NCK – *Numerical Control Kernel*) – núcleo do sistema – interpreta o programa e implementa os algoritmos de interpolação e de controlo de posição;
- Autómato programável (PLC) - encarrega-se do controlo sequencial, como por exemplo, troca de ferramenta, velocidade da árvore, troca da peça de trabalho, do processamento dos sinais I/O e do controlo global da máquina.

Hoje em dia, os fabricantes de CNC fornecem soluções modulares, tanto a nível de *hardware* como *software* (*Open CNC*), permitindo ao fabricante da máquina ferramenta (ou até ao consumidor final) adaptar o CN de uma forma flexível para uma tarefa específica, atualizar ou até mesmo para expandir um sistema existente. Para além do sistema CNC, os fabricantes também disponibilizam todos os componentes necessários desde *drivers*, *encoders* e até servomotores.

2.2 Principais características dos CNC atuais

A título de exemplo são apresentados, de seguida, alguns sistemas de CNC disponibilizados por dois dos principais fabricantes mundiais: Fanuc e Siemens.

A Fanuc afirma-se como líder mundial com 65% de quota de mercado [4], apresentando diferentes séries para utilização em máquinas de precisão e de alta velocidade, com múltiplos

eixos e percursos. Para corresponder a diferentes utilizações, estes modelos combinam uma operação *user-friendly* com níveis de precisão, robustez e eficiência, elevados, compatíveis com todo o tipo de máquina ferramenta, incluindo centros de maquinagem de 5 eixos e tornos com múltiplos eixos. De seguida apresentam-se as principais características destas séries, no que toca a número de eixos controlados e consequente aplicação (Figura 2.1):

- 30i-MODEL B:
 - Número máximo de percursos simultâneos: 10 – 15;
 - Número máximo de eixos controlados: 96 eixos conjugados – 72 eixos de avanço e 24 eixos de *spindle* para 10 percursos e 72 eixos conjugados – 56 eixos de avanço e 16 eixos de *spindle* para 15 percursos;
 - Número máximo de eixos controlados simultaneamente: 24;
 - Indicado para maquinagem com 5 eixos.
- 31i-MODEL B:
 - Número máximo de percursos simultâneos: 6;
 - Número máximo de eixos controlados: 34 eixos conjugados – 26 eixos de avanço e 8 eixos de *spindle*;
 - Número máximo de eixos controlados simultaneamente: 4;
 - Indicado para tornos e centros de maquinagem.
- 31i-MODEL B5:
 - Número máximo de percursos simultâneos: 6;
 - Número máximo de eixos controlados: 34 eixos conjugados – 26 eixos de avanço e 8 eixos de *spindle*;
 - Número máximo de eixos controlados simultaneamente: 5;
 - Indicado para tornos e centros de maquinagem de 5 eixos.
- 32i-MODEL B:
 - Número máximo de percursos simultâneos: 6;
 - Número máximo de eixos controlados: 34 eixos conjugados – 26 eixos de avanço e 8 eixos de *spindle*;
 - Número máximo de eixos controlados simultaneamente: 4;
 - Indicado para tornos e centros de maquinagem;

- 0i-MODEL F:
 - Número máximo de percursos simultâneos: 2;
 - Número máximo de eixos controlados: 34 eixos conjugados – 11 eixos de avanço e 4 eixos de *spindle*;
 - Número máximo de eixos controlados simultaneamente: 4;
 - Indicado para tornos e centros de maquinagem.



Figura 2.1 - Interface do Fanuc 30i, 31i e 32i (à esquerda) e 0i (à direita) [5].

Pontos comuns aos vários sistemas:


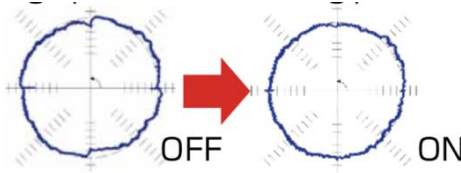
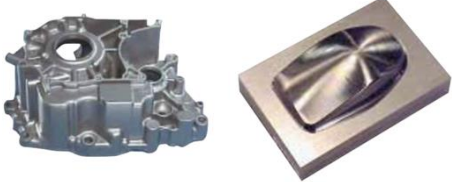
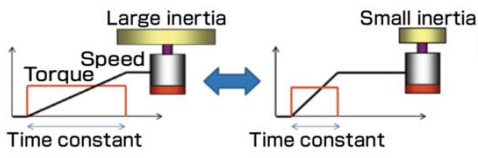
- Redes industriais suportadas: EtherNet/IP, FL-net, DeviceNet, Modbus/TCP, PROFINET, PROFIBUS-DP e CC-Link;
- Interface de comunicação: FSSB (FANUC Serial Servo Bus), FANUC I/O Link *i* para ligação entre sensores e periféricos;
- Ligação PC: Ethernet (100Mbps);
- Consola: *i* Pendant;
- Motorização: FANUC *AC Servo Motor*, FANUC *AC Spindle Motor* e FANUC *Servo Amplifier* compatível com todos os modelos;
- Controlo CNC: função de controlo remoto do CNC através do PC¹, controlo direto do CNC trabalhando na memória e controlo remoto através do *browser*;
- Funções de customização de menus de operação: *C Language Executor* – criação de menus de operação usando linguagem C; *FANUC PICTURE* – criação de menus de operação usando uma filosofia de *pick-and-place* dos vários componentes *e. g.* botões e lâmpadas;
- Função de deteção de avarias com vista a reduzir o tempo de paragem da máquina;

¹ FOCAS - FANUC Open CNC API Specifications

- *Software: NCGuide* – desenvolvimento e *debug* de ecrãs e programas *ladder* no PC; *NCGuide Education Package* – realização de operações de treino no PC sem recorrer ao uso da máquina ferramenta; *Fanuc Program Transfer Tool* – transferência de ficheiros do PC para o CNC *via Ethernet*; *FANUC Servo Guide* – permite a afinação dos vários eixos.

Na Tabela 2.2 estão exemplificadas algumas funções CNC presentes nestes sistemas.

Tabela 2.2 - Funções CNC disponíveis nos vários modelos de controlo numérico do fabricante Fanuc (adaptado de [6])

<i>Função</i>	<i>Ilustração</i>	<i>Descrição</i>
<i>Smart Overlap</i>		Redução do tempo de maquinação através de um desvio da ferramenta a partir de um movimento rápido (G00) para um movimento linear (G01) e vice-versa.
<i>Smart Backlash Compensation</i>		Compensação da folga dos vários componentes mecânicos com vista ao melhoramento da descrição de trajetórias.
<i>AI contour control</i>		Através de uma leitura prévia dos 30 blocos seguintes (<i>look-ahead</i>), esta tecnologia calcula a velocidade ótima de maquinação ² .
<i>Smart Feed Axis Acc/Dec</i>		Ajuste automático das rampas de aceleração/desaceleração dos eixos de movimento para diferentes cargas.

² *AI Contour Control I+/II+* leem até 30 e 200 blocos, respetivamente. Estes valores podem chegar aos 1000 quando ativada uma função de expansão.

Em relação à Siemens foram analisados os modelos incluídos na Figura 2.2.

- SINUMERIK 808D - Modelo correspondente à entrada de gama da série SINUMERIK. Através da sua elevada compacidade e operabilidade é usado em tarefas básicas de torneamento e fresagem. É um controlador de fácil manuseamento e manutenção.
- SINUMERIK 828D - Modelo usado em máquinas ferramenta produzidas em grandes quantidades. Divide-se nos submodelos 828D BASIC, 828D e 828D ADVANCED. Em princípio permite combinar compacidade, custo e elevadas prestações. É usado em aplicações de torneamento e fresagem.
- SINUMERIK 840D sl - Corresponde ao modelo mais poderoso oferecido pela Siemens na secção de controlo numérico, sendo o modelo mais aberto e flexível usado em tarefas de torneamento, fresagem, corte por laser e jato de água, *etc.* É indicado para os utilizadores que necessitam de desenhar e projetar as máquinas ferramenta à medida. Este é apresentado como um CNC aberto, ao nível da MMI e do NCK, em que é possível ao consumidor final criar novas interfaces para a MMI, a partir de linguagens *standard* de programação (C++ e C#).



Figura 2.2 - Várias soluções da Siemens no setor do controlo numérico (adaptado de [7]).

Na Tabela 2.3 pode ser consultada a informação sobre o número de eixos controlados e aplicações dos vários modelos *SINUMERIK*.

Tabela 2.3 - Número e tipo de eixos controlados pelos vários modelos SINUMERIK [8]

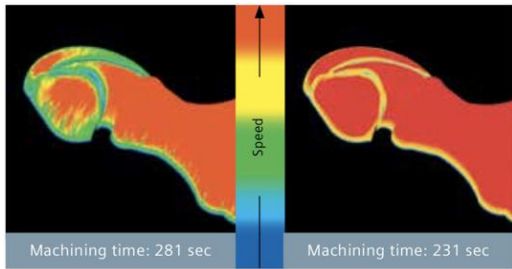
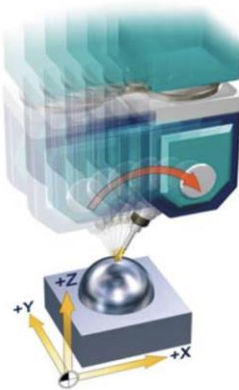
Modelos SINUMERIK	808D	828D	840D sl
Número máximo de percursos simultâneos	1	2	30
Número máximo de eixos controlados conjugados	5	10	93
Número máximo de eixos controlados simultaneamente	4	8	20

Como pontos comuns aos vários sistemas, destacam-se os seguintes:

- Toda a automação reunida no mesmo *software*: *TIA Portal*
- Interface de comunicação: PROFIBUS e PROFINET;
- Ligação PC: Ethernet/RS-232C;
- Consola: *SINUMERIK HT 8*;
- *Software*: *SinuTrain* – simulação, geração e verificação de programas em PC, correspondendo a um ambiente idêntico aos modelos 828D e 840D, *startGUIDE* – ajuda interativa para o modelo 808D;
- Programação gráfica: *Animated Elements*;
- Interface do utilizador baseada em Linux® ou em Windows®, dependendo do modelo.

Na Tabela 2.4 estão exemplificadas algumas funções CNC presentes nestes sistemas.

Tabela 2.4 - Funções CNC disponíveis nos vários modelos de controlo numérico SINUMERIK (adaptado de [9])

<i>Função</i>	<i>Ilustração</i>	<i>Descrição</i>
<i>Advanced Surface</i>		Função <i>look-ahead</i> melhorada que, juntamente com algoritmos de compressão de blocos, permite velocidades de maquinagem superiores e melhor acabamento superficial.
<i>TRAORI</i>		Função exclusiva para maquinagem de 5 eixos <i>e. g.</i> moldes. Permite que a ponta da ferramenta esteja estacionária em relação à peça em movimentos de reorientação.

Da análise dos sistemas CNC destes dois fabricantes pode-se concluir que:

- Ambos têm soluções modulares, o que permite uma maior customização por parte dos utilizadores;
- As funções de CNC presentes nos dois sistemas são muito semelhantes e têm como função primordial, melhorar a exatidão da maquinagem e obter velocidades de maquinagem superiores, através de uma leitura antecipada dos blocos do programa CN;
- A Fanuc tem uma compatibilidade transversal em termos de motorização, enquanto que a Siemens tem uma compatibilidade segmentada.

2.3 Sistemas CAM

Genericamente o *software* CAM permite criar um programa de maquinagem, partindo da definição da geometria e parâmetros de maquinagem das operações a realizar, numa linguagem conhecida como código G. No entanto, apesar de todas as máquinas usarem este tipo de linguagem, existem parâmetros específicos para cada tipo de máquina. De um modo geral, os programas de CAM operam do seguinte modo:

- Definição das formas a obter, usando modelos geométricos internos equivalentes aos usados em sistemas CAD 2D e 3D:
 - Definição da geometria inicial (*stock*) e final da peça a maquinar;
 - Importação do modelo de *software* CAD proprietário.
- Incorporação de informação de maquinagem:
 - Escolha de ferramentas;
 - Definição dos parâmetros de corte (velocidades de avanço e de corte);
 - Escolha e definição de estratégias de maquinagem bem como percursos a efetuar.
- Simulação da execução dos percursos programados:
 - Visualização da peça após cada operação, identificando o excesso de material ou sobre-espessura ainda existente;
 - Detecção de colisões;
 - Cálculo de tempos de maquinagem.
- Configuração para pós-processamento e geração do código no formato específico para a máquina escolhida.

Como ilustração das capacidades atuais destes sistemas apresentam-se de seguida dois exemplos de *software* CAM: Mastercam e NX CAM.

Mastercam

O *software* Mastercam [10] é o CAM mais utilizado no mundo [11] e foi desenvolvido nos EUA pela CNC Software, Inc.. Detém várias funcionalidades que ajudam e possibilitam a criação de programas de maquinagem, tais como:

- Modelação CAD: criação de modelos 2D/3D, superfícies e sólidos;
- Importação de ficheiros em múltiplos formatos *e.g.* SOLIDWORKS com conversor CAD integrado *e. g.* STL;

- Criação de percursos de maquinagem, no caso de fresagem: contornos, caixas com ilhas, furações, letras, *etc.*;
- Geração automática de um código NC aplicado à máquina específica através pós-processadores configuráveis;
- Gestão de operações e simulação dos percursos de maquinagem.

Na Figura 2.3 salientam-se alguns dos módulos da mais recente versão Mastercam 2018.

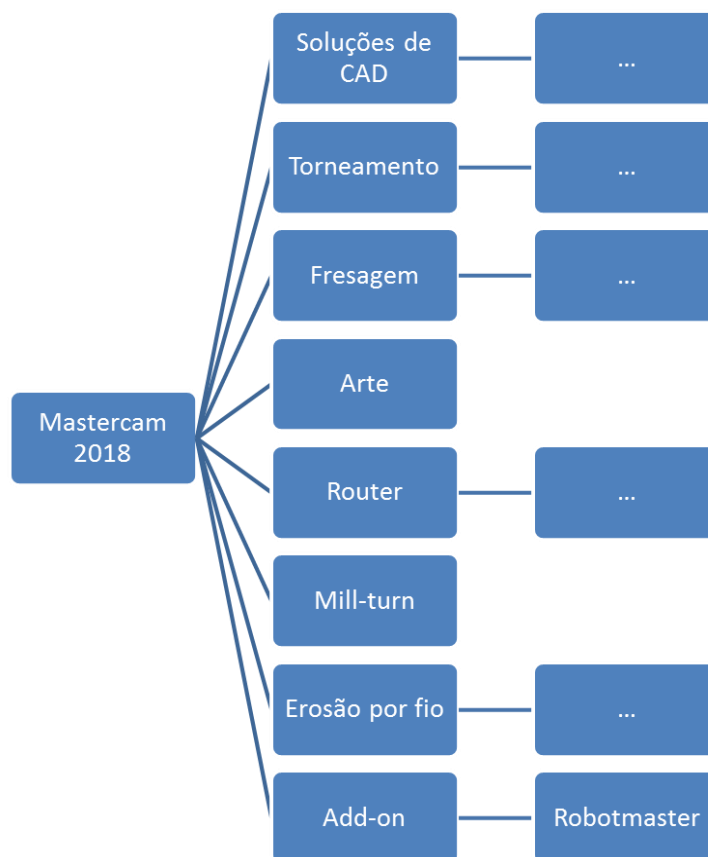
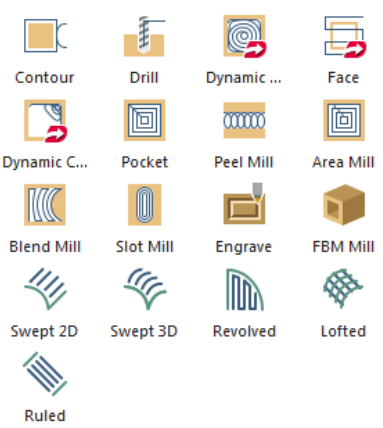
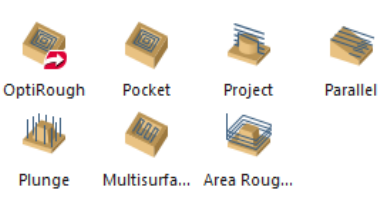





Figura 2.3 - Módulos do Mastercam 2018 (adaptado de [12]).

O Mastercam possui diferentes estratégias de maquinagem que permitem a execução automática de percursos, interiores ou exteriores, podendo estas ser realizadas em desbaste ou acabamento. As estratégias de maquinagem do Mastercam 2018, dividem-se em três grupos: 2D, 3D e *Multi-eixo* e podem ser observadas na Tabela 2.5.

Tabela 2.5 - Estratégias de maquinagem disponibilizadas pelo Mastercam 2018

2D	3D	Multi-eixo
	Desbaste	Padrão
		
	Acabamento	Aplicação
		

NX CAM

Siemens NX é uma ferramenta vocacionada para CAD/CAM/CAE, procurando abranger todo o processo de desenvolvimento de um determinado produto. Este *software* disponibiliza várias aplicações como se pode observar na Figura 2.4.

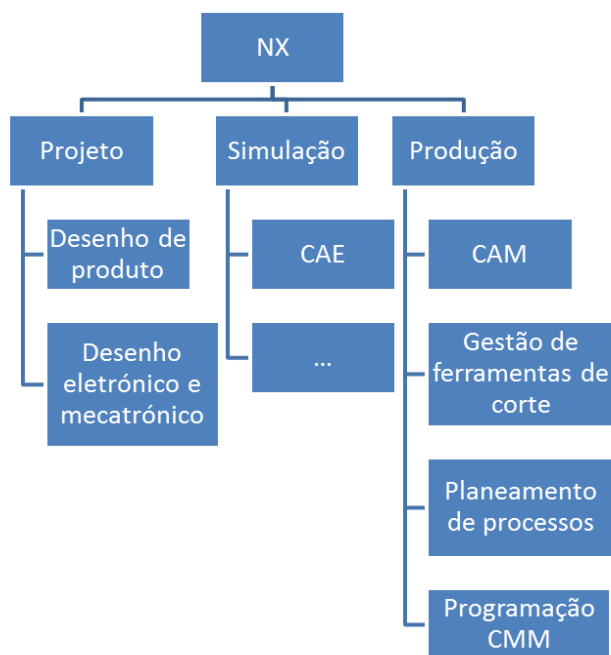


Figura 2.4 - Módulos do NX.

O NX CAM está integrado nas outras soluções do NX, permitindo aos utilizadores o acesso a ferramentas variadas de projeto, montagem e de definição, todos na mesma plataforma.

O módulo CAM deste *software* conta com:

- Fresagem de 2.5/3/5 eixos;
- Funções de maquinagem em alta velocidade;
- Corte por fio;
- Torneamento;
- Biblioteca de pós-processadores e função *PostBuilder* que permite a criação e edição de pós-processadores;
- Maquinagem de hélices e pás de turbinas.

A título de curiosidade, as versões mais recentes do NX contam com um módulo de processo de *additive manufacturing* que permite a elaboração de geometrias complexas (incluindo cavidades), para posterior maquinagem (*hybrid additive manufacturing*).

Com base na informação disponível, referente a cada *software*, é possível identificar as principais funcionalidades:

- As funções de CAD oferecidas pelos dois sistemas são compatíveis com a importação e manipulação de modelos de CAD, ao nível do *software* de CAD mais evoluído;
- O NX integra-se numa aplicação PLM (*Product Lifecycle Management*) sendo mais abrangente nas funcionalidades oferecidas, desde o projeto até à produção de um produto, enquanto o Mastercam limita-se à geração de programas de CN;
- Ambas as plataformas permitem, com *software* adicional, a programação de robôs industriais, Robotmaster no Mastercam e Robotics Machining no NX, a fim de realizar tarefas de maquinagem, seja fresagem, lixagem, rebarbagem ou corte por laser.

3 Maquinagem com robôs industriais

Neste capítulo são listadas as várias aplicações de robôs industriais enfatizando as que envolvem maquinagem. Nesse sentido, realiza-se uma comparação entre os robôs e as máquinas CNC, apresentando-se alguns exemplos de robôs especializados em maquinagem e ainda se descrevem alguns casos de aplicações existentes no mercado. Finalmente, é feito um levantamento dos vários modos de programação dos robôs, onde é apresentado *software* relevante na área, com ênfase nas tarefas de maquinagem.

3.1 Aplicações de robôs industriais

Segundo a norma ISO 8373:2012, um robô industrial é um manipulador controlado automaticamente, reprogramável e multifuncional com três ou mais eixos que pode estar fixo num lugar ou a uma plataforma móvel para efeitos industriais [13].

Para além do controlador e do atuador, uma configuração típica de um robô inclui também uma consola de programação (Figura 3.1).

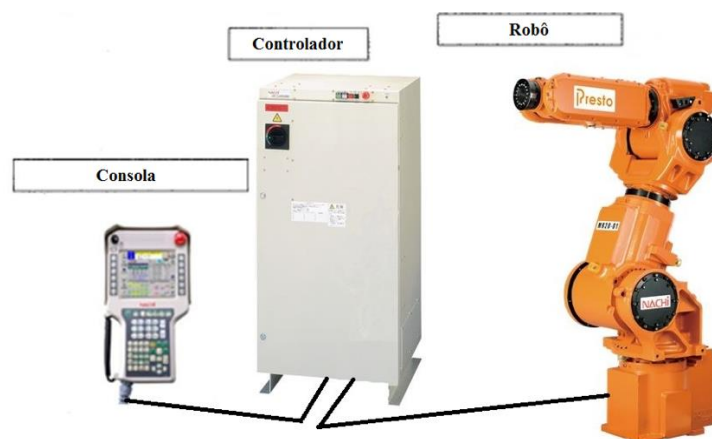


Figura 3.1 - Configuração típica de um robô [14].

No panorama industrial, os robôs são usados nomeadamente na manipulação de materiais, soldadura, pintura, polimento ou até tarefas de maquinagem. Na Figura 3.2 está representada a distribuição relativa de vendas de novos robôs, por área de aplicação industrial referentes ao ano de 2016.

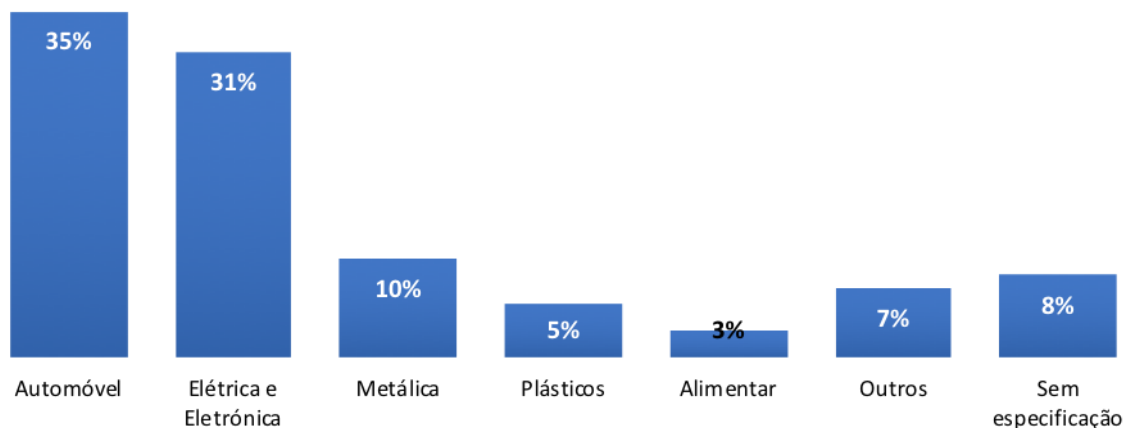


Figura 3.2 - Distribuição dos robôs vendidos em 2016 por setor industrial [15].

Com base na análise do gráfico, é notória a baixa utilização dos robôs em aplicações metálicas, onde se inclui em operações de maquinagem, facto que é corroborado por Chen e Dong [10], que realça que o número de robôs vendidos na área de maquinagem foi de 5% face à venda total de robôs nesta aplicação. Acresce ainda salientar que, há que ter em conta que as operações de maquinagem realizadas são normalmente de polimento e rebarbagem, pelo que a fresagem é uma aplicação de pequena dimensão. A utilização dos robôs, aplicados na maquinagem a nível industrial, é evidenciada na Tabela 3.1.

Tabela 3.1 - Exemplos de aplicações de maquinagem usando robôs [16]

<i>Indústria</i>	<i>Processo</i>	<i>Produto</i>
Todas as aplicações	Fresagem	Prototipagem rápida
Aeroespacial	Lixagem, polimento, furação e corte	Pás de turbinas, isolamentos e fuselagem
Automóvel	Lixagem, furação, fresagem e corte	Motores, chassis, carroçaria, puxadores de porta e para-choques
Moda	Fresagem e lixagem	Moldes de manequins e manequins
Fundição	Rebarbagem, fresagem, furação e acabamento	Peças fundidas e moldes
Naval	Fresagem	Cascos de barcos
Médica	Lixagem e polimento	Próteses
Entretenimento	Fresagem	Cenários
Plásticos	Fresagem, <i>routing</i> ³	Moldes e capacetes
Madeira	Fresagem	Mobiliário, corrimões e ornamentação

Os manipuladores robóticos possuem uma maior flexibilidade, permitindo maquinar peças de maiores dimensões, atingindo a mesma posição usando diferentes configurações dos seus eixos. Este aspeto representa uma grande vantagem quando se trata de maquinagem de peças com geometrias complexas e de difícil acesso. No entanto, a maquinagem com robôs tem a limitação de não permitir obter valores de tolerância dimensional e acabamento superficial, na ordem de grandeza de uma máquina de CNC convencional, o que pode originar desvios na ordem dos 10 mm [17]. A Tabela 3.2 resume as diferenças mais notórias entre uma máquina CNC e um robô.

³ Máquina utilizada para realizar trabalhos manuais em madeira, designada Tupia em Português.

Tabela 3.2 - Robôs vs máquinas CNC nas aplicações de maquinagem (adaptado de [18])

<i>Parâmetro</i>	<i>Máquina CNC</i>	<i>Robô</i>
Exatidão	~ 0.005 mm	~ 0.1 – 1.0 mm
Repetibilidade	~ 0.002 mm	~ 0.03 – 0.3 mm
Espaço de trabalho	Limitado	Grande
Extensão do espaço de trabalho	Impossível	Possível
Configuração física	Cartesiana (3 eixos) + 2 eixos rotativos	Normalmente de revoluta
Nº de eixos	3 a 5	6 +
Rigidez Estrutural	Elevada	Reduzida
Propriedades dinâmicas	Homogêneas no espaço de trabalho	Heterogêneas no espaço de trabalho
Linguagem de programação	Baseada em Código G&M	Específica do fabricante (KRL, RAPID, etc.)
Flexibilidade	Operações semelhantes	Qualquer tipo de operação
Custo	Alto	Baixo

Atualmente, não existe qualquer entrave quando o material a maquinar é de baixa dureza e.g. espumas, madeira, plástico ou cera, independentemente do estágio de produção: protótipo, produto final ou intermédio. No entanto, os problemas despontam quando se pretende maquinar bronze, cobre, ligas de alumínio ou ferro fundido.

A capacidade dos robôs de levar a cabo operações de maquinagem em 3, 4 ou 5 eixos, a alta flexibilidade no posicionamento e orientação da ferramenta, a sua área ou volume de trabalho e o custo de uma célula robótica, tornam este sistema de produção numa alternativa competitiva face a uma máquina CNC, quando a elevada tolerância dimensional das peças a obter não é um requisito fundamental.

Assim sendo, as barreiras a ultrapassar no uso de robôs na maquinagem estão relacionadas com a baixa exatidão, originadas pelas forças de corte e da vibração do sistema derivado à sua baixa rigidez. Logo, a rigidez do robô é o fator mais limitador quando a dureza

do material aumenta e se pretende taxas de remoção de material elevadas [16], [19], [18], [20].

Os mais recentes avanços tecnológicos têm demonstrado uma preocupação em ultrapassar as barreiras que dificultam a aplicação de robôs articulados na área de maquinagem. Neste contexto, enumeram-se alguns pontos que condicionam a utilização de robôs em operações de maquinagem:

- *Software* – A falta de uma norma para a programação de robôs, como existe nas máquinas CNC, é uma limitação no desenvolvimento de soluções de *software*. Dentro do *software* disponível para maquinagem com robôs destacam-se os seguintes: RobotWare Machining FC da ABB, Robotguide da Fanuc, CAMRob da Kuka, MotoSim EG da Motoman, Robotmaster para o Mastercam, PowerMILL Robot Interface da Autodesk, NX CAM Robotics ou Sprutcam;
- Exatidão e rigidez – O desenvolvimento de robôs para aplicações de maquinagem é ainda reduzido. Porém, os fabricantes têm alguns modelos adequados para esta aplicação: TX200 HSM da Stäubli, IRB 7600 da ABB, KR 600 F da Kuka, MH80 II da Yaskawa e o F200iB da Fanuc.

Resumidamente, seria ousado garantir que a maquinagem com robôs substituirá as máquinas CNC. Por um lado, o uso do CNC prende-se com a qualificação do processo e, consequentemente, é mais dispendioso. Por outro lado, a robótica é cada vez mais vista como uma alternativa viável, quando o cumprimento de tolerâncias não é especialmente exigido. Atualmente existem robôs utilizados em maquinagem, tais como os exemplificados a seguir na Tabela 3.3.

Tabela 3.3 - Exemplos de robôs para operações de maquinagem

Stäubli TX200 HSM [21]	
<ul style="list-style-type: none"> . Cargas até 150 kg . Alcance de 2194 mm . Repetibilidade de ± 0.06 mm . Vários tipos de maquinagem: acabamento, furação, rebarbagem e contornos . Estrutura protegida (poeira, lubrificante, etc.) . Mudança automática da ferramenta . Controlador baseado em tecnologia CNC (não requer conhecimento da programação de robôs) . HMI permite controlar os parâmetros do <i>spindle</i> 	
ABB IRB 6660 [22]	
<ul style="list-style-type: none"> . Até 205 kg para um alcance de 1.93 m . Repetibilidade de 0.07 – 0.11 mm . Orientado para operações de maquinagem: furação, fresagem, rebarbagem, acabamento e contornos . Proteção IP 67 e proteção dedicada à apara . <i>Software</i> dedicado RobotStudio Machining PowerPac . RobotWare Machining Force Control para uma força de contacto controlada indicada para aplicações de polimento 	
YASKAWA MH80 II [23]	
<ul style="list-style-type: none"> . Cargas até 80 kg . Alcance de 2061 mm . Repetibilidade de ± 0.07 mm . Possibilidade de ser montado em várias posições diferentes . Elevada compacidade . Manuseamento, maquinagem e montagem 	

KUKA KR 600 R2830 F [24]	
<ul style="list-style-type: none"> . Cargas até 1300 kg . Alcance de 2826 mm . Repetibilidade de ± 0.08 mm . Manuseamento, paletização, maquinagem e montagem . Desbaste e tarefas menos exigentes: furação, lixagem e rebarbagem . HMI intuitiva da SINUMERIK . Controlador baseado em tecnologia CNC: KUKA.CNC 	

Sendo vários os exemplos de robôs aptos para operações em maquinagem, são referenciadas, de seguida, algumas empresas que recorrem a robôs nos seus projetos.

Fundilusa

Empresa sediada em Vila Nova de Cerveira. A sua área de trabalho abrange a fundição de cobre e respetivas ligas para produção de hélices de alta qualidade para barcos. Os robôs usados pertencem ao fabricante KUKA usando o *software* SprutCAM. Estes são usados em aplicações de acabamento de propulsores (Figura 3.3) e no desbaste de protótipos em madeira.

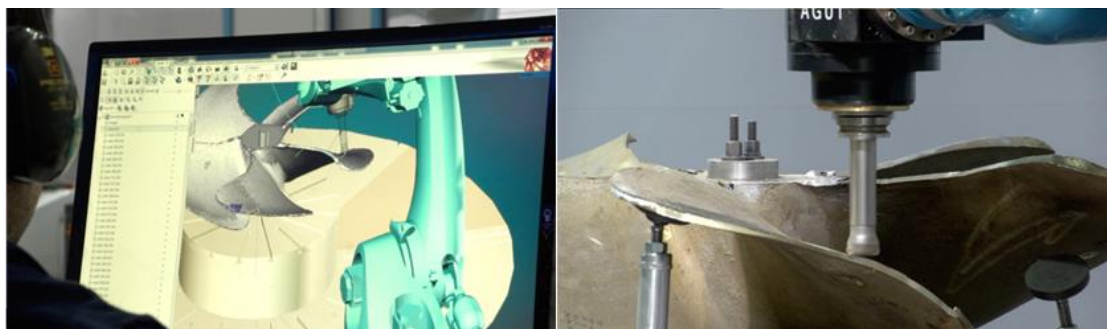


Figura 3.3 - Simulação no *software* SprutCAM (à esquerda) e operação de acabamento (à direita) [25].

AV & R

Automation Vision and Robotics [26] é uma empresa Canadiana líder em automação tendo o seu foco em soluções de automação, visão e robôs. Grande parte do seu negócio envolve a indústria aeroespacial e energética. Nestes campos, a equipa é conhecida pelas operações de acabamento recorrendo a robôs (polimento, rebarbagem e lixagem) e inspeção visual automática com o objetivo de produzir soluções “feitas à medida”.

A AV & R é especialista na rebarbagem de câmaras de combustão para naves espaciais. Este processo é extremamente complexo, podendo envolver trajetórias muito detalhadas. Sem o uso de *software* específico, foram gastas 900 horas na programação do robô recorrendo a uma equipa de cinco engenheiros. Com o uso do Robotmaster esse tempo diminui em 75% usando apenas uma equipa de dois engenheiros [27].

Na Figura 3.4 é mostrado um robô do fabricante Staübli a realizar uma operação de polimento. À esquerda encontram-se as várias lixas, enquanto que, no canto superior direito existe um sistema de inspeção visual.

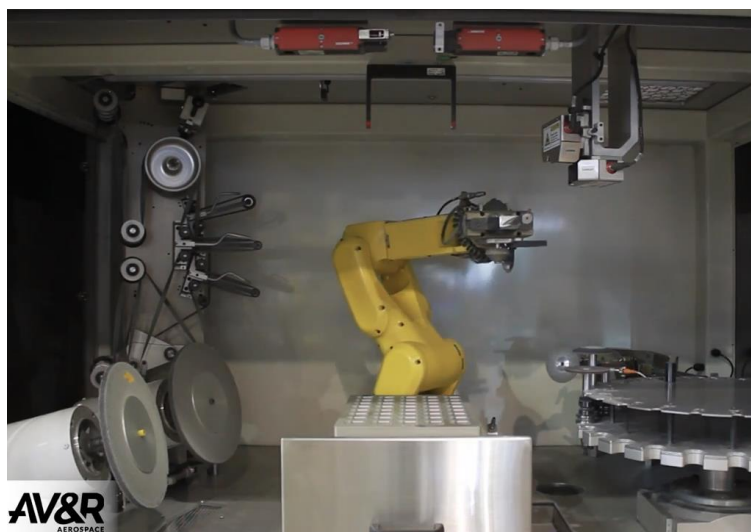


Figura 3.4 - Solução robótica de polimento de pás de turbinas a gás [28].

Johnson Wellfield Quarries

A empresa JWQ [29] tem um histórico de trabalhar a pedra que remonta há 150 anos. O seu mais recente avanço tecnológico foi a aquisição de uma célula robótica, constituída por robôs antropomórficos da Staübli auxiliados pelo *software* Mastercam/Robotmaster.

Um dos seus trabalhos mais famosos foi a decoração da praça Tudor, Sheffield, Inglaterra. O processo partiu de maquetes 1:10 das esculturas sendo, posteriormente, foram alvo de um processo de ampliação recorrendo a uma combinação de análise laser 3D e

modelação digital, com vista à criação de modelos em tamanho real, a partir dos quais os robôs esculpiram as peças finais (Figura 3.5).

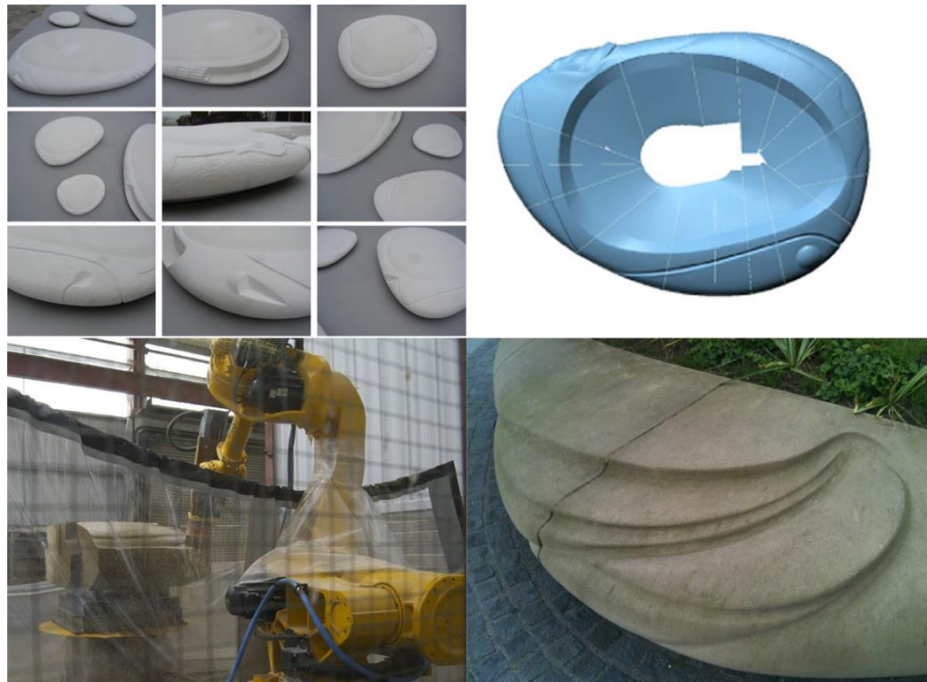


Figura 3.5 - Processo de criação de esculturas [30].

3.2 Programação de robôs

A programação de robôs pode ser feita de dois modos diferenciados: *online* ou *offline*. Seguidamente será feito o paralelismo entre eles, salientando-se o que os distingue e o que os aproxima.

Programação *online*

Este tipo de programação envolve a definição das localizações a atingir pelo robô através do seu guiamento pelo operador, sendo estas gravadas no controlador do mesmo. Através destas localizações é possível a definição de percursos. A responsabilidade no posicionamento e na orientação do robô, recai sobre o operador que tem de garantir que não ocorrem colisões durante a movimentação do robô [31]. Neste âmbito, podem distinguir-se dois modos de programação (Figura 3.6):

- Guiamento – pode ser direto ou assistido, no qual o operador/robô está encarregue de mover diretamente o robô;
- Consola de programação – Movimentação indireta do robô através da consola de programação.

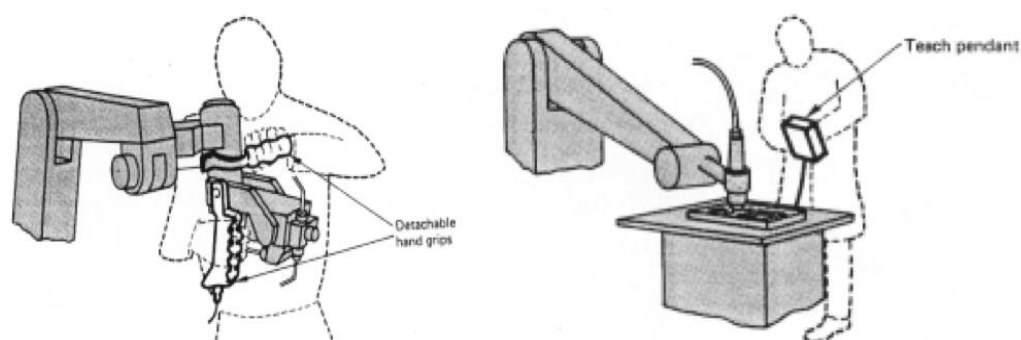


Figura 3.6 - Programação por guiamento (à esquerda) e programação via consola (à direita) [32].

Relativamente ao método *online*, apesar de parecer simples, é um processo em que o robô não pode ser utilizado enquanto é programado, possuindo pouca flexibilidade, na medida em que, na eventualidade de alteração do programa, é necessário repetir o processo outra vez. Tendo em conta os entraves enunciados, este método tem aplicabilidade reduzida quando se pretende usar o robô em operações de maquinação, exceto em operações de acabamento superficial.

Programação *offline*

A programação *offline* consiste na criação e simulação de programas recorrendo a *software* dedicado. Para além disso, recorrendo a este método os programas tornam-se mais flexíveis, na medida que, podem ser alterados a qualquer momento. Assim, é possível a verificação e validação dos movimentos do robô, evitando possíveis colisões na célula real. Na Tabela 3.4 são apresentadas algumas ferramentas de programação de robôs existentes no mercado.

Tabela 3.4 - *Software* proprietário de programação e simulação *offline* de robôs

Fabricante	<i>Software</i>
ABB	RobotStudio
Kuka	KUKA SIM
Motoman	MotoSim EG
Fanuc	ROBOGUIDE

Para além deste *software* disponibilizado pelos fabricantes de robôs existem outras aplicações que permitem a programação de RI de diferentes fabricantes na mesma plataforma (ver Tabela 3.5).

Tabela 3.5 - *Software* de programação genérico de robôs

<i>Software</i> genérico	Robôs suportados (exemplos)
Robcad	ABB, Adept, Kuka, Fanuc, Motoman, Stäubli
Delmia	
CimStation Robotics	
Workspace 5	
Octopuz	
ArtiMinds RPS	

A programação de RI, para operações específicas de maquinagem, pode ser realizada tirando partido de *software* de CAM, dotado de pós-processadores integrados na aplicação (Tabela 3.6) ou recorrendo a aplicações informáticas para pós-processar os programas gerados em *software* CAM genérico (Tabela 3.7).

Tabela 3.6 - *Software* CAM com pós-processador integrado

<i>Software</i>	Robôs suportados (exemplos)
SprutCAM	ABB, Kuka, Fanuc, Motoman, Stäubli
NX CAM	
Mastercam/Robotmaster	
Powermill Robot	
WorkNC Robot	

Tabela 3.7 - *Software* de pós-processamento

<i>Software</i> CAM	<i>Software</i> de Pós-processamento	Robôs compatíveis
Mastercam, NX CAM, CATIA, SURFCAM, TopSolid, Creo, PowerMill, WorkNC, Cimatron, SolidCam, Alphacam, VISI, Edgecam, GO2cam, CAMWorks, GibbsCAM, Tebis	RoboDK	ABB, Adept, Comau, Denso, Fanuc, GSK, HIWIN, Kairo, Kuka, Kawasaki, Mecademic, Mitsubishi, Motoman, Nachi, OTC, Panasonic, Precise, Stäubli, Toshiba, Universal Robots, Yamaha
	IRBCAM	ABB, Adept, Comau, Denso, Hyundai, Fanuc, KUKA, Mitsubishi, Motoman, Stäubli, Toshiba, Universal Robots
	Eureka	ABB, Kuka, Fanuc, Motoman, Kawasaki, Staubli, Comau
	G-Code Converter EG	Motoman
	CAM Converter	ABB
	KUKA.CAMRob	Kuka

Neste sentido, procede-se à apresentação de duas aplicações de programação de robôs usadas neste trabalho.

Robotmaster

O Robotmaster é uma extensão do *software* Mastercam que permite a programação de diferentes robôs industriais. Dentro da panóplia de processos compatíveis, destacam-se alguns [33]:

- Fresagem;
- Rebarbagem;
- Soldadura;

- Polimento;
- Manuseamento;
- Retificação;
- Pintura;
- Tratamento superficial;
- Processos aditivos.

A nova versão deste *software* possui uma tecnologia, denominada de RISE, que representa o ambiente interativo do Robotmaster, onde é possível visualizar e otimizar em tempo real eventos, tais como, colisões, singularidades e limitações de alcance. Quaisquer alterações efetuadas na movimentação são rapidamente recalculadas e visualizadas.

Na Figura 3.7 está representado o ambiente de simulação; neste destacam-se o espaço de trabalho do robô (a roxo), a posição da peça a maquinar, referencial intermédio, bem como as trajetórias de maquinagem (a azul claro). Adicionalmente, é possível observar a posição do TCP do robô, dado pelo referencial à direita, bem como o referencial representante da base do manipulador, à esquerda.

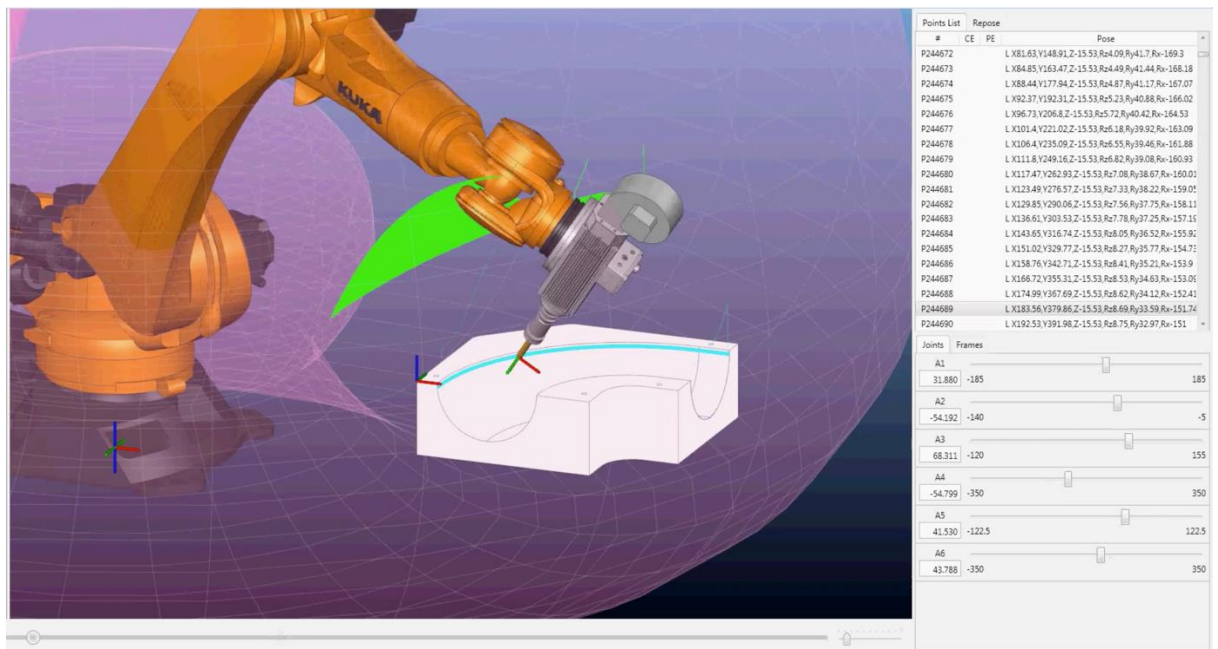


Figura 3.7 - Ambiente de simulação do Robotmaster [34].

Outra funcionalidade a referir é a *Optimization* onde é possível ajustar a posição da ferramenta em relação à peça de trabalho. Na opção básica deste é possível visualizar-se um ecrã onde podem ser sinalizados vários problemas recorrentes da movimentação do robô (ver Figura 3.8). A região amarela significa os limites das juntas, a região roxa significa “*wrist*

flip” e a região azul significa fora de alcance. Esta funcionalidade também oferece opções avançadas, onde é possível limitar os valores a atingir pelas juntas.

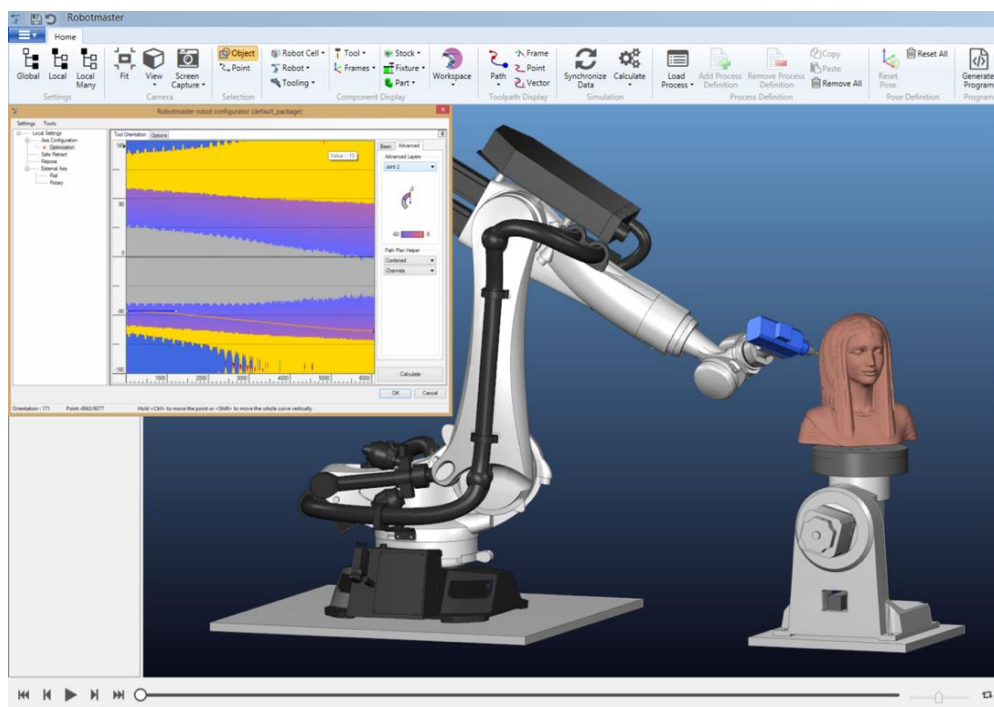


Figura 3.8 - Funcionalidades avançadas do menu *Optimization* [35].

RobotStudio/Machining PowerPac

O RobotStudio é o *software* para programação *offline* e simulação dedicado exclusivamente aos robôs da ABB. Este inclui uma réplica do controlador real (*RobotWare*) do robô, o que permite a realização de simulações realistas. O Machining PowerPac é uma extensão do RobotStudio orientada para operações de maquinagem leve, como por exemplo, retificação, polimento ou rebarbagem.

A utilização deste *software* permite:

- Definição de trajetórias:
 - Seleção da aresta ou face a maquinar;
 - Definição de parâmetros de maquinagem;
 - Definição de estratégias de maquinagem.
- Conversor CAM – Conversão de código G para a linguagem RAPID do robô;
- Otimização e edição da trajetória – Análise de trajetória para a eventualidade de colisão, singularidade, etc.;

- Simulação e verificação das trajetórias geradas;
- Possibilidade de importar modelos CAD ou um programa CNC;
- Possibilidade de criação de ferramentas.

Com base na informação recolhida, referente a cada *software*, é possível identificar as principais funcionalidades de cada um:

- Ambos permitem a criação de geometrias e sólidos, e ainda importação de diferentes formatos CAD;
- O RobotStudio é vocacionado para operações de acabamento, por isso, não contém ferramentas na sua biblioteca para operações de fresagem, no entanto, é possível a criação e parametrização dessas ferramentas e também programação de robôs para operações de fresagem, ainda que de uma forma pouco expedita, comparativamente com o Robotmaster;
- O Mastercam possui vários pós-processadores para diferentes modelos de robôs, ao invés do RobotStudio, que só permite a programação de robôs da marca ABB;
- Ambos possuem modos de simulação com capacidade de deteção de colisões, no entanto, no RobotStudio não existe uma representação da peça final após a operação de maquinagem.

4 Características dos sistemas utilizados

Neste capítulo, apresentam-se as principais características da mesa de dois eixos (XY) e do respetivo controlador (CNC), bem como as características da célula robótica. Nessa caracterização incluem-se os sistemas de suporte à elaboração e simulação/teste dos respetivos programas de maquinagem, as possibilidades destes sistemas implementarem os testes idealizados e recolherem a informação necessária para a respetiva avaliação.

4.1 Elementos principais da mesa XY e controlador CNC

O controlador CNC da mesa XY recorre a um sistema modular da Fagor, modelo 8055M integrado na bancada de suporte visível na Figura 4.1. A bancada integra o teclado do CNC, módulos do sistema (unidade central, eixos, disco rígido e ligação à rede), um monitor, fonte de alimentação, três *drivers* e ainda todos os componentes eletromecânicos de suporte à alimentação elétrica e comando do sistema.



Figura 4.1 - Sistema Fagor 8055M.

A Unidade Central (UC) corresponde ao centro de processamento do sistema e integra as funções de CNC, nas quais se destacam a capacidade de interpretação de programas, funções de interpolação e controlo de posição dos eixos de movimento. Também contempla um *softPLC* que permite lidar com todas as funções auxiliares, como por exemplo, refrigeração, equipamentos de segurança e controlo global da máquina. O controlador possui uma construção modular, permitindo ao fabricante ou ao utilizador escolher os módulos e configuração que melhor se adequam às suas necessidades.

Este sistema é um modelo relativamente antigo e foi configurado integralmente, há cerca de três anos, não tendo sido reconfigurado para este presente trabalho. A versão de *software* do CNC instalada é a 11.20, de 2008.

Módulos dos eixos – Sistema Fagor DDS

O sistema DDS foi criado pela Fagor e é usado no controlo de servomotores, especialmente concebidos para acionamento de eixos de movimento ou da árvore de máquinas ferramenta. A estrutura de controlo dos servomotores síncronos (*Brushless AC*), usados no acionamento da mesa XY, recorre ao controlo de posição na unidade central do CNC e o controlo de velocidade e corrente nos *drivers* (Figura 4.2). O controlo de velocidade e de corrente utiliza controladores PI e a comunicação entre a UC e os *drivers* faz-se por intermédio da rede SERCOS. Os ganhos destes controladores podem ser alterados consoante a aplicação através do *software* WinDDSSetup.

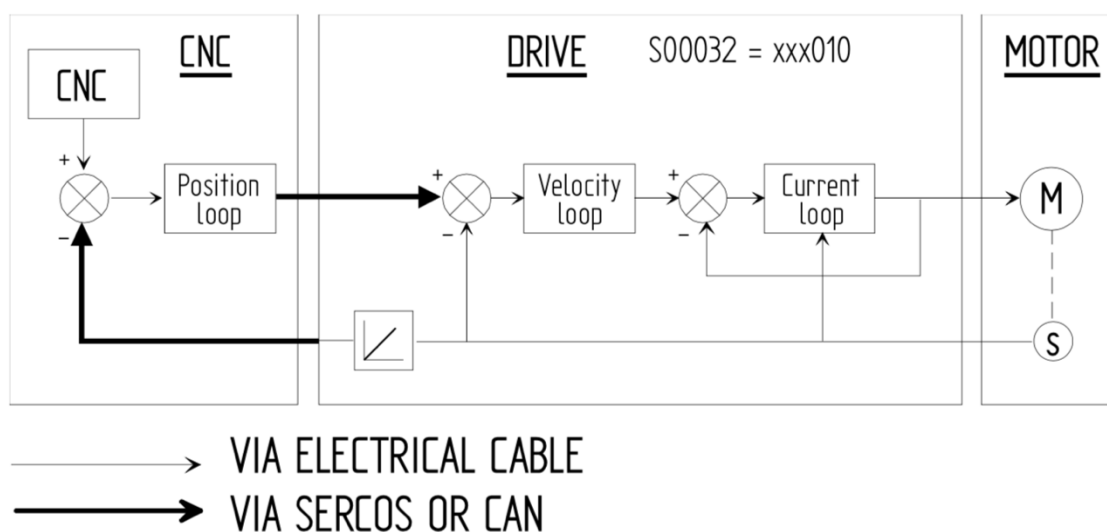


Figura 4.2 - Driver de velocidade com feedback do motor [36].

Relativamente ao sistema de acionamento existem três servomotores com *encoder* de referência (FXM31.40A.E1) de 1.1 kW, com uma velocidade nominal de 4000 .

Estrutura da mesa XY

A mesa XY é constituída por dois eixos lineares, usando para cada eixo um sistema de transmissão de fuso e porca de esferas, de modo a converter o movimento angular dos servomotores em movimento linear (Figura 4.3). O curso máximo de cada eixo é de 511 mm e a mesa de trabalho (600x600mm) é suportada por guias de secção circular e rolamentos lineares. Tendo em conta as limitações dos sistemas de transmissão, servomotores e respetivos *drivers*, as características de carga e movimentos da mesa foram configuradas para os seguintes valores [37]:

- Velocidade máxima linear programável de 15000 mm/min;
- Aceleração máxima – $0,5 \text{ m/s}^2$ – limitada pelo tempo de aceleração de 500 ms ACCTIME, definido nos parâmetros máquina.

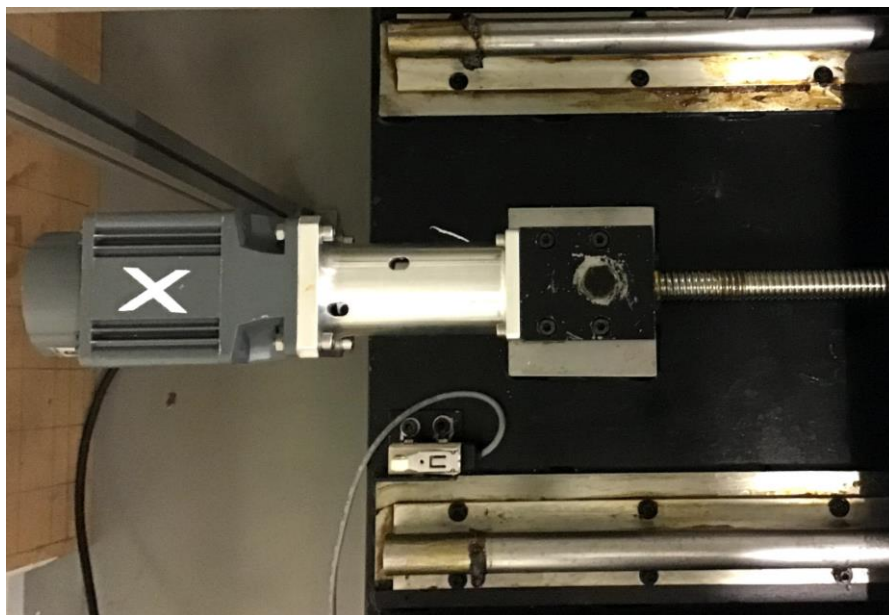


Figura 4.3 - Eixo linear do movimento X (servomotor, fuso e guias de apoio).

Osciloscópio (*Oscilloscope*)

A função osciloscópio é uma ferramenta de suporte do CNC Fagor que permite representar, em tempo real, a evolução de sinais, definidos como variáveis do CNC ao longo do tempo. Estas variáveis podem ser manipuladas num menu tendo como objetivo o ajuste adequado dos parâmetros do CNC. Esta função é particularmente importante neste trabalho, pois foi através dela que foi monitorizada a variação da velocidade de avanço, nos vários percursos implementados e é constituída por vários menus (Figura 4.4):

- *Scale e offset* – Variação da amplitude do sinal, movimento vertical e horizontal ou ajuste da base de tempo para todos os sinais;
- *Analysis* – Analisa ponto-a-ponto cada sinal usando dois cursores;
- *Parameter* – Atribuição de novos valores aos parâmetros máquina do CNC, definidos no menu Configuration;
- *Configuration* – Definição das variáveis a analisar, estabelecimento do número de amostras, tempo de amostragem e modo de captura do sinal:
 - Definição das variáveis através da cor de representação e o código associado (Tabela 4.1);
 - O tempo de amostragem tem de ser um valor múltiplo de 4 e toma um valor mínimo de 4 ms;
 - O número de amostras varia entre 1 e 1024;
 - O modo de captura pode ser simples, em que o CNC para de monitorizar a variável, quando atinge o número de amostras especificado, ou contínuo, quando o sinal preenche o ecrã.
- *Begin* – Captura os sinais com base na configuração estabelecida previamente.



Figura 4.4 - Função *Oscilloscope* disponível no CNC.

Tabela 4.1 - Exemplos de variáveis que podem ser atribuídas a um dos quatro canais do *Oscilloscope*

<i>Variável</i>	<i>Descrição</i>
FREAL	Velocidade de avanço
FREAL(X-C)	Velocidade de avanço do eixo
FTEO(X-C)	Velocidade de avanço teórica do eixo
DRPO(X-C)	Posição indicado pelo <i>driver</i> do eixo

A função *Oscilloscope* tem uma limitação significativa, que passa por não permitir guardar ou enviar para o PC o valor das variáveis. Uma possível alternativa seria usar o *software* WinDDSSetup, onde já é possível guardar esses sinais. No entanto, apenas admite a monitorização de um eixo de cada vez, o que impossibilita, por exemplo, a monitorização da velocidade de avanço na descrição de um movimento circular, em que existe um movimento conjugado dos dois eixos. Assim, optou-se por fazer um registo manual dos sinais gerados no osciloscópio do CNC, já que este representa o método mais completo e flexível. O tempo de amostragem escolhido foi de 24 ms, tal como no robô.

Teste Ballbar

Em teoria, se se programasse o CNC para realizar um movimento circular e o desempenho fosse perfeita, então o círculo realizado seria uma cópia exata do que foi programado. Na prática, muitos fatores relativos à estrutura da máquina, sistema de transmissão e sistema de controlo, podem conduzir o CNC a executar um círculo, que difere, em raio e forma, face ao círculo programado. Ou seja, o teste Ballbar corresponde a uma função de ajuste dos eixos e consiste na programação e verificação de um círculo, como o indicado no gráfico da Figura 4.5. O seguinte exemplo mostra o programa usado para a execução de um círculo de 20 mm de diâmetro (Figura 4.5).

```

N00 G01 G05 F1200 X150.000 Y150.000
N10 G02 X150.000 Y150.000 I20.000 J0.000
N20 (GOTO N00)

```

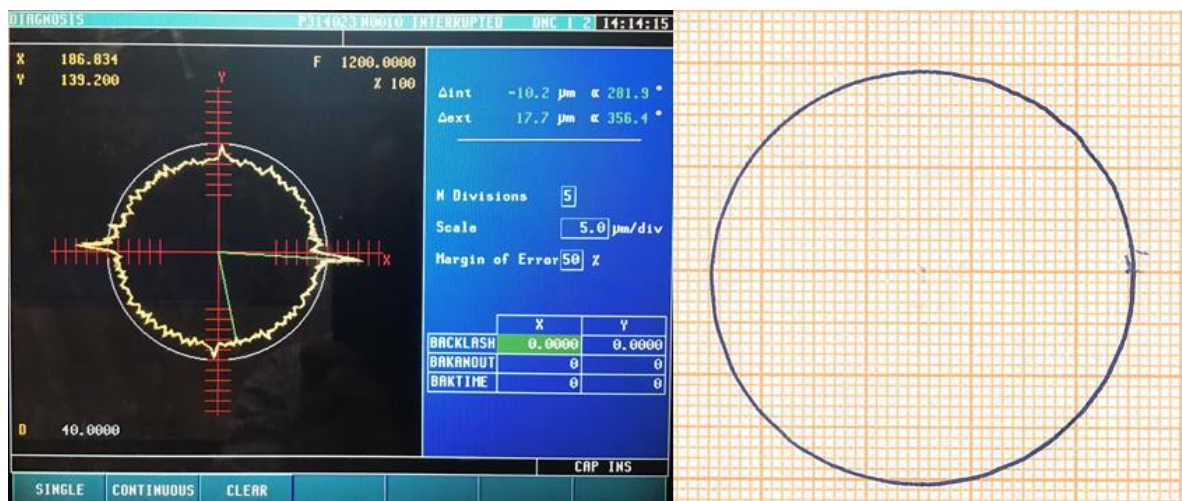


Figura 4.5 - Teste ballbar realizado para uma velocidade de 1200 mm/min (à esquerda) e o círculo obtido (à direita).

Este teste foi realizado no plano XY, para uma velocidade de avanço de 1200 mm/min, onde o máximo desvio registado foi de 17,7 μm @ 356,4° e o mínimo desvio foi de 10,2 μm @ 281,3°. De notar que os picos de desvio foram detetados na inversão do movimento do eixo X. Estes correspondem ao erro de movimento da máquina e podem dever-se a falhas do mecanismo de transmissão de movimento como o motor, *driver*, fuso de esferas, guia linear ou rolamentos. Estes componentes devem ser verificados e reparados ou substituídos, de modo a restaurar a exatidão do eixo [38].

4.2 Funcionalidades de configuração e programação do CNC 8055M

As alternativas de programação são comuns às normalmente encontradas nos sistemas CNC, incluindo programação em códigos G e numa linguagem de alto nível. A lista completa de instruções G encontra-se no anexo A.

A estrutura dos programas de CN consiste numa sequência de blocos de programação executados sucessivamente. Cada bloco contém um conjunto de informação definida através de códigos particulares, como é ilustrado no seguinte exemplo do manual de programação da *Fagor* [39]:

N22 G01 X30 Y40 Z40 F10000 GOTO N11 ; Não executar

- **Início do bloco:** é opcional e pode ser formada por uma ou várias condições de salto e pelo número do bloco - **N22**;
- **Bloco do programa:** contempla os comandos em linguagem ISO ou linguagem de alto nível:
 - Linguagem ISO: vocacionada para o controlo dos eixos de movimento e eixo principal de corte:
 - Funções preparatórias (G), múltiplas instruções com diferentes funções que condicionam a execução de movimentos. No caso de funções de interpolação, apenas disponibiliza as funções clássicas de interpolação linear (G00 e G01) e circular (G02 e G03);
 - Funções para especificação da velocidade de avanço (F) e velocidade de rotação da árvore (S);
 - Funções de identificação das ferramentas (T) e respetivas dimensões a usar para compensação automática de ferramentas, no percurso implementado;
 - Códigos M destinados a várias funções auxiliares ou complementares, tais como, controlo de sistemas ON/OFF ou automatismos sequenciais (*e. g. ativar spindle, parar programa, etc.*).
 - Linguagem de alto nível: permite aceder a variáveis de uso geral, tabelas e variáveis do sistema. Podem ser usadas expressões do tipo IF, GOTO, CALL, *etc.*, que se assemelham a outras linguagens de programação. Também permite usar qualquer tipo de expressão aritmética, de relação ou lógica e possui ainda, instruções para a construção de ciclos, como também sub-rotinas com variáveis locais.
- **Final do bloco:** é opcional e poderá conter o número de repetições do bloco e/ou comentários sobre o mesmo.

Tendo em conta os códigos ISO do Anexo A identificaram-se as funções mais relevantes na comparação com as funções disponibilizadas pelo controlador do RI, na execução de trajetórias envolvendo variações de velocidade e posição, e no controlo de

trajetória. Em primeira instância são pormenorizadas as funções importantes no controlo de trajetória, em seguida, as funções particulares relacionadas com o controlo de velocidade e posição na execução de cantos e ainda na sincronização de movimentos em blocos de programação sucessivos.

Trajectoria circular tangente à trajetória anterior (G08)

Através desta função é possível a programação de uma trajetória circular tangente à trajetória anterior, sem necessidade de programar as coordenadas do centro. Para a definição da trajetória circular é apenas necessário a especificação do ponto final. É de salientar que esta função não é modal e, como tal, não é memorizada sendo necessário a sua programação sempre que se desejar uma trajetória tangencial à anterior. De seguida são apresentados os blocos de programação correspondentes ao uso da função G08, ilustrados na Figura 4.6.

G90 G01 X70 Y40

G08 X90 Y60

G08 X110 Y60

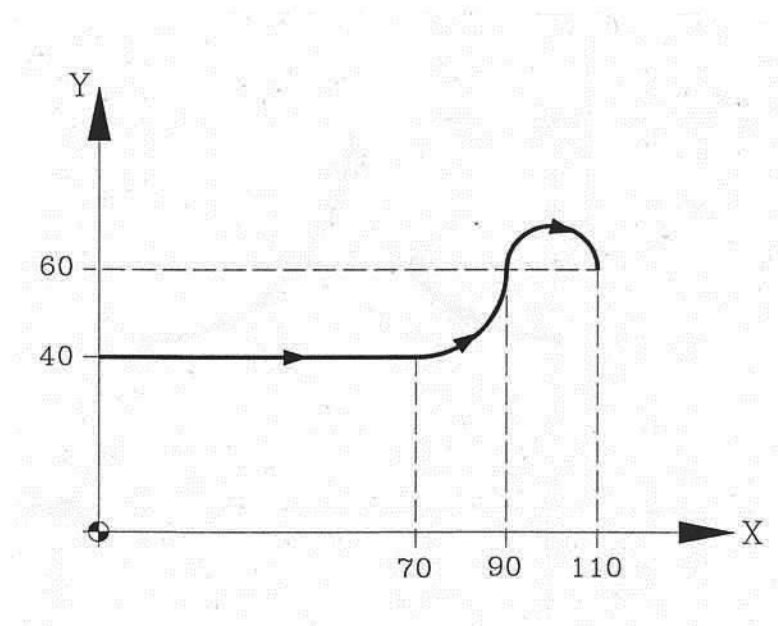


Figura 4.6 - Trajetória circular tangente à trajetória (G08) [40].

Trajectoria circular definida mediante três pontos (G09)

Usando a função G09 é possível definir uma trajetória circular (arco), através da definição do ponto intermédio e final, em que o ponto inicial do arco é o ponto de partida do movimento. Ou seja, em vez de se programar as coordenadas do centro, programa-se qualquer

ponto intermédio. De seguida é apresentado o bloco de programação correspondente ao uso da função G09, ilustrada na Figura 4.7.

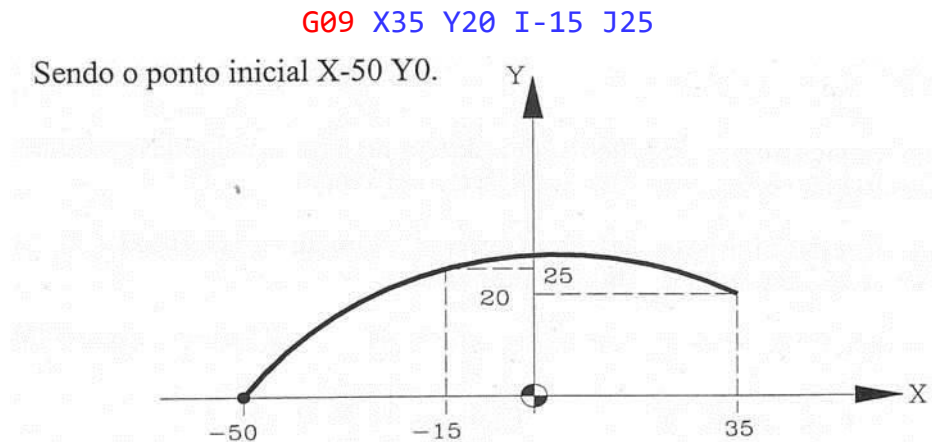


Figura 4.7 - Trajetória circular definida mediante três pontos (G09).

Interpolação helicoidal (G02/G03)

A interpolação helicoidal deriva de uma interpolação circular no plano de trabalho e do deslocamento do resto dos eixos programados. Seguidamente, é apresentado o bloco de programação correspondente à interpolação helicoidal ilustrada na Figura 4.8.

G03 X0 Y0 I15 J0 Z50 K5

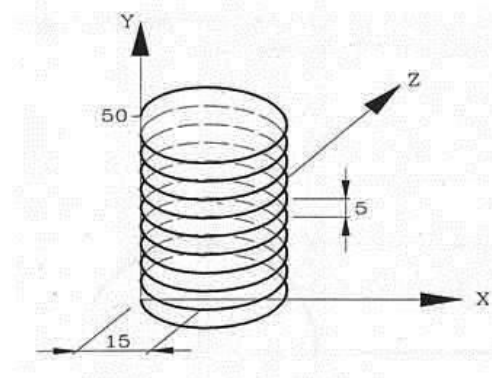


Figura 4.8 - Interpolação helicoidal.

Aresta viva (G07)

É uma função modal em que o CNC não começa a executar o bloco seguinte do programa, até que seja atingida a posição programada. Esta, considera-se alcançada quando o eixo se encontra a uma distância inferior a *INPOSW* (*dead band*), da posição programada.

Este parâmetro tem um valor pré-definido de 0.01 mm, configurável nos parâmetros máquina. Com esta função pretende-se garantir que o percurso programado (teórico) coincida, tanto quanto possível, com o percurso real, como se pode observar na Figura 4.9.

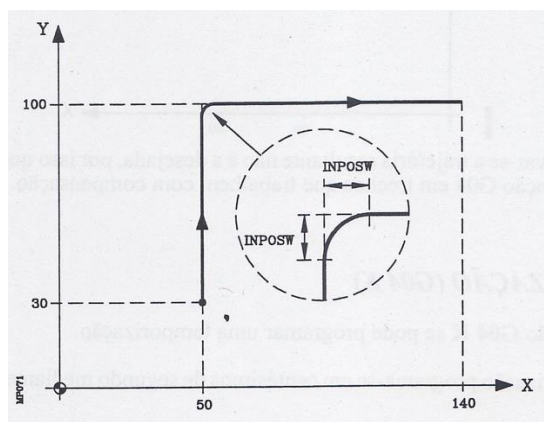


Figura 4.9 - Função de aresta viva (G07) [40].

Arredondamento de aresta controlado (G50)

É uma função modal em que o CNC, depois de finalizar a interpolação teórica do bloco atual, espera que seja atingida a posição dentro da zona *INPOSW2*, para continuar com a execução do bloco seguinte. Este parâmetro tem um valor pré-definido de 0.01 mm e pode ser configurado nos parâmetros máquina. Esta função impõe que a diferença entre o perfil teórico e o real, seja inferior à definida no parâmetro *INPOSW2* (Figura 4.10).

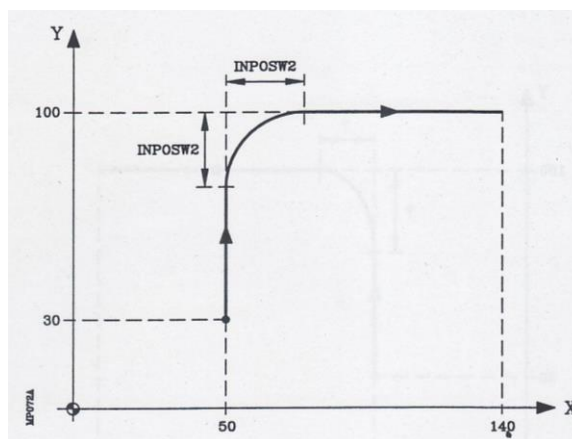


Figura 4.10 - Arredondamento de aresta controlado (G50) [40].

Arredondamento de aresta (G05)

Também é uma função modal, em que o CNC começa a execução do bloco seguinte do programa, depois de finalizada a interpolação teórica do bloco atual, não esperando que os eixos estejam na posição programada. As diferenças entre o perfil teórico e real, ao contrário

do arredondamento de arestas controlado, dependem do valor da velocidade de avanço programada (Figura 4.11), assim, é expetável que quanto maior for a velocidade de avanço, maior será a diferença entre ambos os perfis.

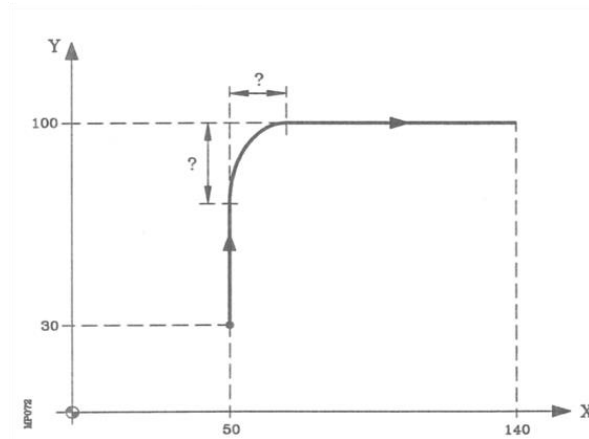


Figura 4.11 - Função de arredondamento de aresta G05 [40].

Look-ahead (G51)

É uma função que permite minimizar a redução de velocidade (Figura 4.12) linear entre blocos, com deslocamentos muito pequenos entre si. Neste âmbito, é possível atingir uma velocidade de maquinagem mais alta, reduzindo os tempos de execução do programa em questão [2]. O formato de programação contém um parâmetro E, que corresponde ao erro de contorno, expresso em milímetros. Quanto menor for este valor, menor será a velocidade de avanço resultante.

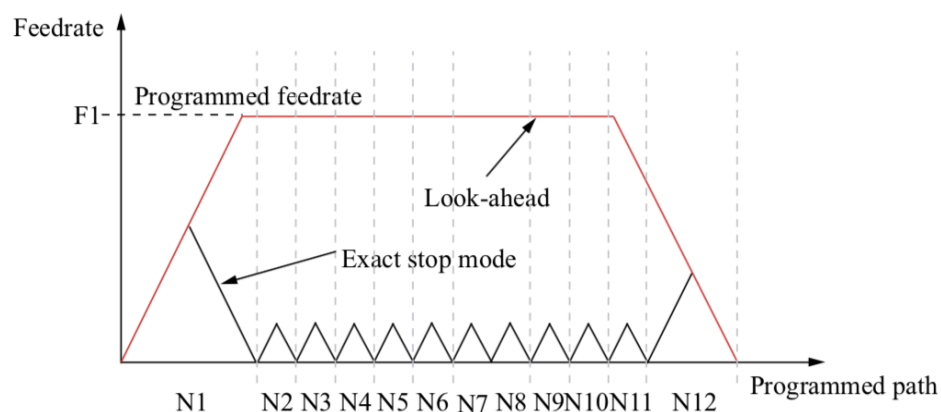


Figura 4.12 - Efeito da função *look-ahead* na velocidade de avanço (adaptado de [2]).

Este CNC particular analisa antecipadamente os 50 blocos à frente do bloco atual, calculando a velocidade máxima linear em cada intervalo de blocos.

Foram realizados alguns testes preliminares que mostraram que, apenas as funções arredondamento de aresta (G05) e *look-ahead* (G51), revelaram trajetórias reais diferentes das trajetórias programadas, o que levou à sua escolha para serem comparadas com outras funções de movimento do robô.

4.3 Funcionalidades de programação dos movimentos do robô IRB 2400

A célula robótica real é constituída pelos seguintes elementos:

- Robô ABB IRB 2400;
- Controlador ABB IRC5;
- Mesa ABB IRPB C-500;
- *Spindle* (XLC70) e respetivo controlador (não utilizados);
- Elemento terminal constituído pela célula de carga (ATI Delta IP60) e sistema pneumático de mudança rápida de movimento (SCHUNK SWS – 011).

Na Figura 4.13 encontra-se representado o elemento terminal do robô, com o conjunto de acessórios e ferramentas acoplados que permitem usar uma caneta para desenhar em papel, os percursos programados, correspondentes ao ponto central de uma ferramenta.

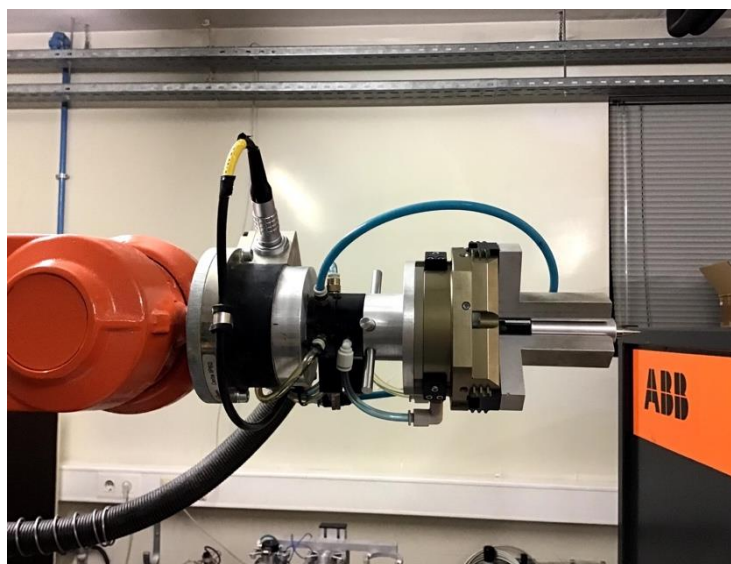


Figura 4.13 - Elemento terminal do robô.

De modo a conseguir uma maior eficiência e realismo na simulação dos processos de maquinação, foi usado o *software* RobotStudio. Neste foi necessário configurar e criar uma célula robótica virtual, com base na célula real de maneira a analisar os casos de estudo. Depois de implementados, procede-se ao envio dos vários programas das trajetórias para o

controlador e estes podem ser editados, usando o PC ou a consola de programação. A programação do robô é feita através da linguagem RAPID, que é proprietária da ABB. As trajetórias executadas podem ser analisadas pelos percursos traçados com a caneta ou pelo *software* RobotStudio com a função *Signal Analyser*. Esta, à semelhança da função *Oscilloscope* do CNC, é uma função que ajuda na otimização dos programas do robô através de uma análise, visualização e captura de sinais provenientes do controlador do robô *e. g.* velocidade do TCP, posição do TCP em relação a determinado eixo, *etc.* O tempo de amostragem dos sinais é de 24 ms, pelo que foi este o valor que foi usado com a função *Oscilloscope* do CNC.

As instruções básicas da linguagem RAPID, no controlo de movimento, incluem instruções do tipo das seguintes:

MoveL P1,v200,fine,caneta\WObj:=newRef;

MoveL P2,v200,z20,caneta\WObj:=newRef;

Onde é possível especificar:

- O tipo de trajetória:
 - Movimento de Juntas – *MoveJ* – implica o movimento independente das juntas do robô de modo a que, quando a posição de destino for atingida, a paragem de movimento dos eixos ocorra no mesmo instante;
 - Movimento Linear - *MoveL* – o movimento do TCP descreve uma trajetória linear (Figura 4.14);
 - Movimento Circular – *MoveC* – o movimento do TCP descreve uma trajetória circular.
- O ponto de destino (P1 e P2) – ponto de paragem ou de passagem;
- A velocidade (*v200*) e aceleração do TCP;
- *Zone* (*z20*), especifica a distância ao ponto intermédio, a partir do qual a trajetória pode ser alterada para atingir a posição seguinte;
- Ferramenta usada pelo robô (caneta), nos movimentos, definindo o TCP e o referencial de trabalho (*newRef*).

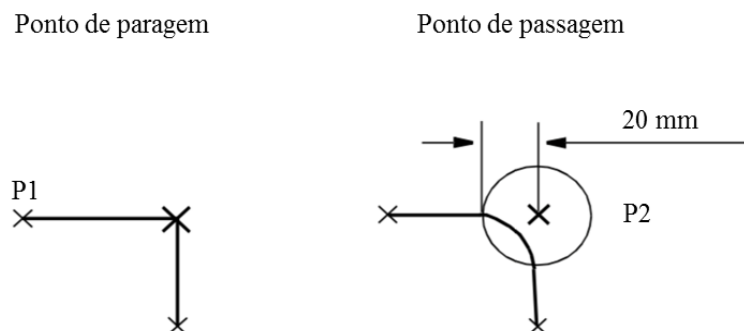


Figura 4.14 - Evidência do parâmetro *zone* na descrição de trajetórias (adaptado [41]).

Os parâmetros de velocidade e de *zone* têm de ser seleccionados na programação em Mastercam/Robotmaster, optando-se por valores pré-definidos ou valores especificados pelo utilizador.

Adicionalmente, também é possível proceder ao ajuste da aceleração do movimento de juntas com o intuito de se obterem movimentos mais suaves. Esta instrução (*AccSet*) permite uma diminuição da aceleração e desaceleração, aconselhada para um manuseamento de objetos frágeis ou em aplicações de maquinagem onde existe preocupação com as vibrações. Este ajuste compreende dois parâmetros, *Acc* e *Ramp* (Figura 4.15). O primeiro parâmetro diz respeito à percentagem de aceleração e desaceleração face ao seu valor máximo. O segundo parâmetro define a taxa de variação da aceleração e desaceleração face aos seus valores normais.

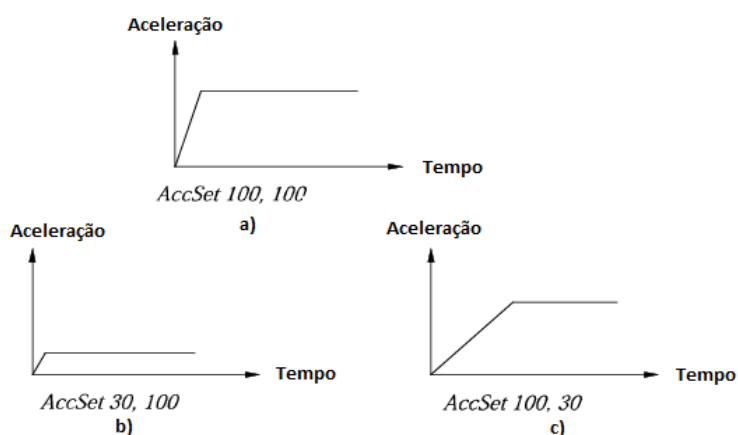


Figura 4.15 - Instrução *AccSet* [41].

Para verificar como esta função afeta os perfis de velocidade foi criado em ambiente de simulação do RobotStudio, um exemplo de um percurso linear definido com 3 pontos. Todas as trajetórias são lineares e as posições dos pontos dos cantos têm uma *zonedata fine*. Na Figura 4.16 está representada a estação de trabalho virtual e o percurso definido (indicado a amarelo).

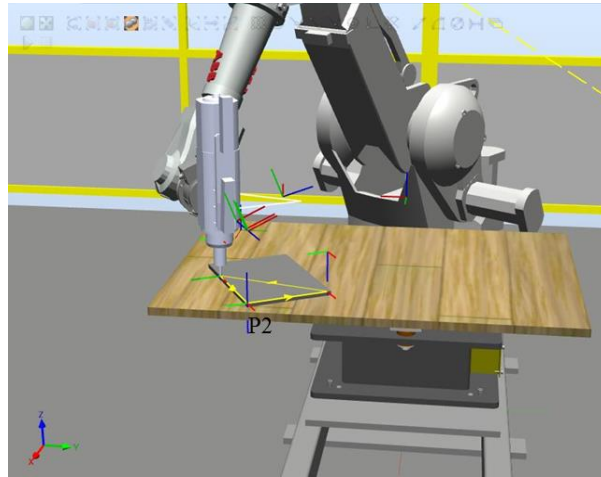


Figura 4.16 - Célula robótica virtual utilizada nos casos de estudo.

Na Figura 4.17 apresentam-se os perfis de velocidade do TCP na zona envolvente do canto onde se pode observar que, quando o TCP atinge o canto (P2), a velocidade é zero, pois é um ponto de paragem. Além disso, note-se a diferença entre dois casos extremos de *AccSet 10, 10* e *AccSet 100, 100*, onde é evidente a diferença de tempo necessário para atingir a velocidade programada.

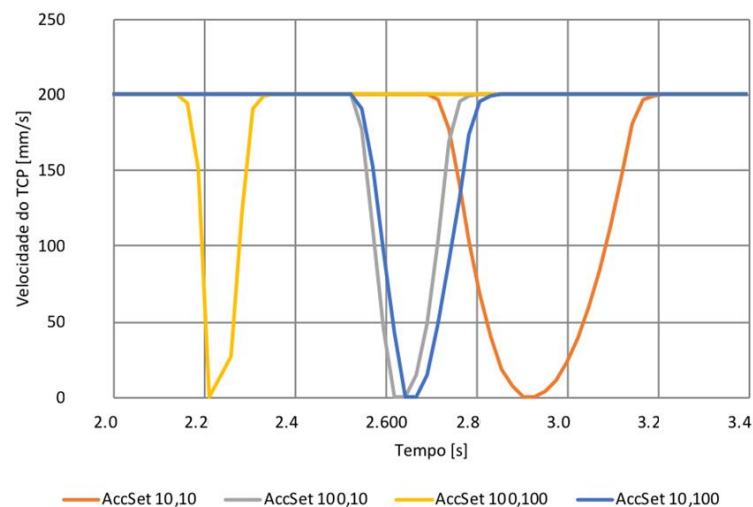


Figura 4.17 - Perfil de velocidades do TCP para as várias parametrizações de *AccSet*.

4.4 Síntese

Com base na análise comparativa dos sistemas utilizados (CNC e robô industrial) foram identificadas as principais funcionalidades de cada um. Na Tabela 4.2 são apresentadas algumas funções, para assinalar diferenças entre os dois sistemas, na programação de trajetórias.

Tabela 4.2 - Modo de execução de trajetórias no sistema CNC e no robô

<i>Trajectoria</i>	<i>CNC</i>	<i>RI</i>
Arco de circunferência	Trajectoria circular mediante três pontos (G09)	MoveC
Movimento linear	Movimento rápido (G00) e interpolação linear (G01)	MoveL
Círculo	Interpolação circular (G02 e G03)	Duas instruções MoveC
Interpolação helicoidal	Interpolação helicoidal (G02 e G03)	Não existe função equivalente
Trajectoria circular tangencial à anterior	Trajectoria circular tangente à trajetória anterior (G08)	Não existe função equivalente

Uma das diferenças encontradas no CNC em relação ao RI, consiste na existência de funções modais no CNC, o que se traduz numa simplificação da programação, nomeadamente em relação à utilização de funções de interpolação. No caso dos movimentos programados no robô é necessário especificar o tipo de interpolação, velocidades e tolerância, associados em cada instrução de movimentação do programa. Por outro lado, no controlador do robô, os valores, limitação de aceleração e de velocidade, podem ser definidos a nível do programa, enquanto no CNC, são configurados a nível dos parâmetros da máquina.

Tendo identificado os parâmetros, que podem ser usados quer no RI quer no CNC, para determinar as condições em que um determinado percurso pode ser executado pelo elemento terminal, foram definidos casos de estudo, apresentados no capítulo seguinte, de modo a estabelecer uma comparação entre estes dois processos de maquinagem tendo em conta que:

- Os movimentos são executados no plano (maquinagem 2D), pois o CNC está limitado ao plano XY;
- A caracterização do comportamento em percursos elementares, ou seja, geometricamente mais simples (segmentos retos e circulares) permitem identificar melhor a contribuição de cada parâmetro na execução dos movimentos;
- A análise de padrões de movimento mais complexos permitirá avaliar em que medida se podem extrapolar os resultados obtidos, na análise de movimentos elementares.

5 Casos de estudo e testes realizados

Este capítulo tem como objetivo apresentar os casos de estudo usados para a comparação dos dois sistemas, na execução de trajetórias compostas por movimentos simples, lineares e circulares, e alguns movimentos complexos comuns em estratégias de maquinagem. Para além das diferentes geometrias consideradas, o número de casos de estudo resulta dos diferentes parâmetros a serem avaliados e incluem: tipo de interpolação, condições de aceleração e velocidades definidas, funções particulares na programação CNC (G05 e G51) e na programação do robô (*zone* e *AccSet*). As trajetórias implementadas foram executadas sobre folhas de papel milimétrico A4, usando como ferramenta uma caneta no elemento terminal do robô e da mesa XY.

5.1 Procedimento e metodologia

Sistema RI

Nos casos de estudo mais simples, apenas foi usado o RobotStudio para a criação das trajetórias, bem como a sua simulação e respetiva exportação dos dados para posterior análise. Num estágio mais avançado, o uso do Mastercam foi imprescindível na criação de várias trajetórias. Aqui foi necessário definir a geometria ou contorno da peça e seleção de uma máquina genérica representativa da família de robôs *ABB*, para a definição dos percursos de maquinagem. Transpondo para o Robotmaster foi necessário proceder à configuração local e global, a fim de escolher o modelo de robô, definir a posição da peça em relação ao robô e configurar os movimentos dos eixos. Ainda neste *add-on* é possível realizar uma primeira simulação, com vista a verificar e validar a posição do referencial, posição e orientação do TCP ou até identificar possíveis erros de movimentação, antes de ser gerado o código RAPID, que será validado e testado no *software* RobotStudio. Uma vez no RobotStudio, é possível fazer uma segunda simulação podendo esta ser analisada usando a função *Signal Analyser*. Depois de validado o código, este é enviado para o controlador do robô onde é executado. Com este processo pretende-se uma análise comparativa da influência da parametrização (*zone*, velocidade do TCP e *AccSet*) e programação em ambiente *offline*

(Mastercam/Robotmaster/RobotStudio), nos percursos descritos pelo robô. O método de trabalho realizado com o robô segue o procedimento ilustrado na Figura 5.1.

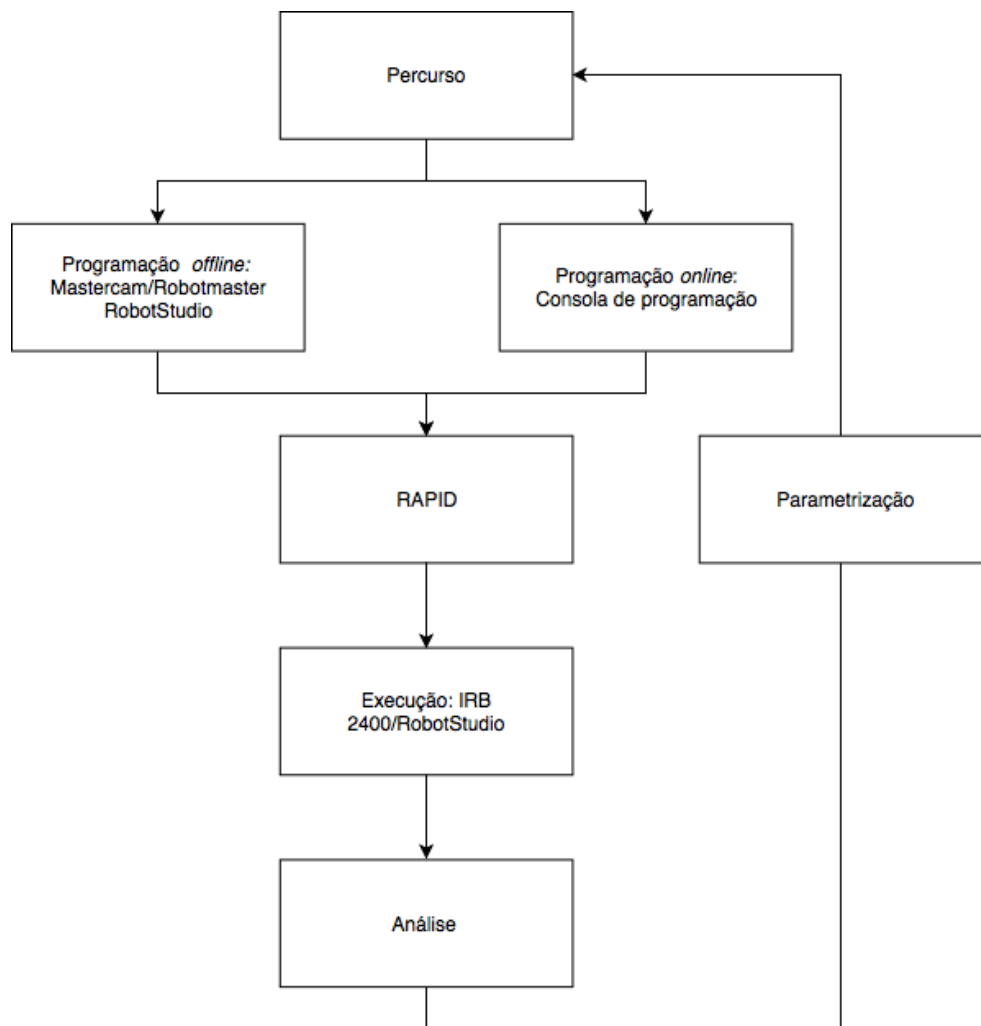


Figura 5.1 - Análise dos casos de estudo do robô.

Sistema CNC

Aqui a programação pode ser feita em computador através de editor de texto, sendo depois enviada para o CNC através do *software* WinDNC, de forma direta na interface do CNC (para os casos simples) ou ainda recorrendo ao Mastercam. Neste último é necessário definir a geometria ou contorno da peça na interface gráfica, escolher uma máquina genérica, seguido da escolha das estratégias de maquinagem e ajuste dos respetivos parâmetros de maquinagem. Posteriormente, executando na máquina o código G proveniente das várias possibilidades de programação, é analisada a trajetória através da função *Oscilloscope* (perfil de velocidade e tempo de execução). Também é possível simular a trajetória no CNC para detetar eventuais erros de posicionamento ou de programação. Assim, a parametrização prende-se com a escolha da velocidade de avanço e funções de controlo de trajetória. Mais uma vez, a Figura 5.2 traduz o procedimento realizado no sistema CNC.

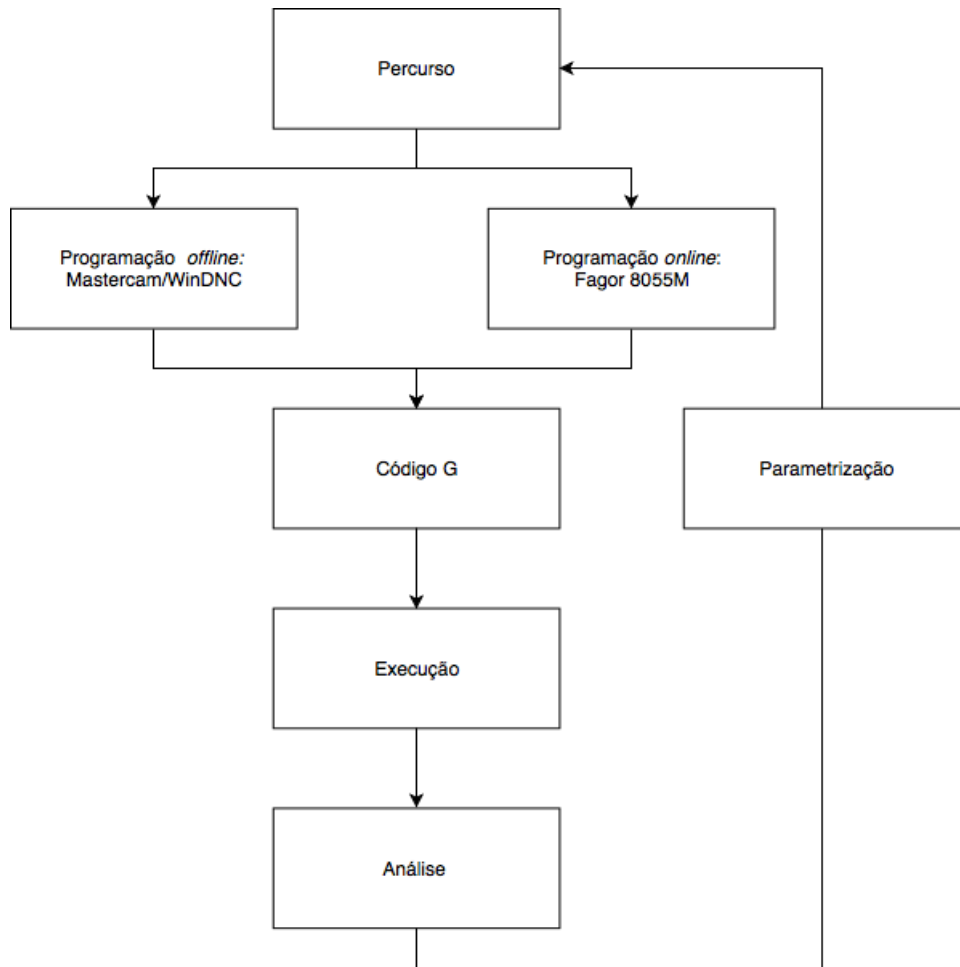


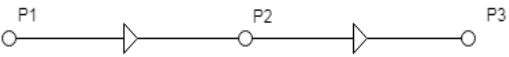
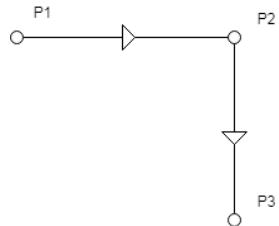
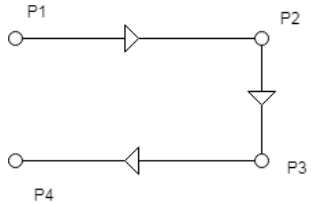
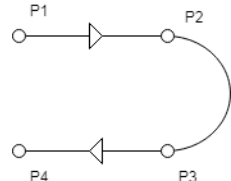
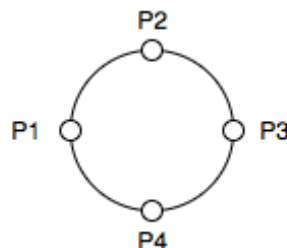
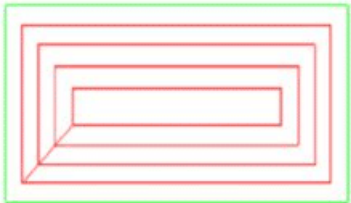
Figura 5.2 - Análise dos casos de estudo do CNC.

A avaliação das consequências das várias parametrizações nos programas gerados, foi realizada em termos de:

- Medidas quantitativas:
 - Tempo de execução do percurso;
 - Perfis de velocidade (variação de velocidade em relação ao seu valor programado);
 - Aceleração;
 - Desvios relativamente ao percurso teórico.
- Medidas qualitativas:
 - Facilidade de programação (extensão de programas);
 - Alternativas de programação;
 - Desvios relativamente ao percurso teórico (forma).

Tendo em conta a geometria associada aos percursos, definiram-se casos de estudo que são apresentados na Tabela 5.1.

Tabela 5.1 - Casos de estudo analisados

<i>Movimentos</i>	<i>Percursos</i>
1. Movimento linear com 3 pontos	
b) linha	
a) canto	
2. Movimento em U	
a) Base reta	
b) Base circular	
3. Movimento circular	
4. Estratégias de maquinagem	

Escolha de parâmetros

No caso do robô foram avaliadas as implicações do parâmetro *zone*, da velocidade e aceleração, assim como das estratégias de maquinagem. Já no caso do CNC avaliaram-se as implicações da velocidade de avanço, da função arredondamento de aresta (G05), da função *look-ahead* (G51) e das estratégias de maquinagem. Na Tabela 5.2 estão apresentados os parâmetros avaliados, em cada caso de estudo, bem como o número de testes realizados.

Visto existir semelhança no tipo de especificação dos parâmetros, *zone* e *look-ahead*, deu-se importância aos seguintes valores comuns: **1 mm**, **5 mm** e **10 mm**. Para o parâmetro *zone* foram escolhidos: **z1**, **z5**, **z10**, **z30** e **z100** e para a função *look-ahead* foram escolhidos os erros de contorno: **E0.1**, **E1.0**, **E2.0**, **E5.0** e **E10.0**. Adicionalmente foram introduzidas outras parametrizações que serviram para mostrar a evolução das trajetórias para casos limite.

Tabela 5.2 - Parâmetros específicos para cada caso de estudo

Parâmetros	Sistema		Casos de estudo			
	CNC	RI	1	2	3	4
Velocidade de avanço	X	X	X	X	X	
<i>Zone</i>		X	X	X	X	
Arredondamento de aresta (G05)	X		X	X	X	
<i>Look-ahead</i> (G51)	X		X	X	X	
Estratégia de maquinagem	X	X				X
<i>AccSet</i>		X			X	
Número de testes (Total de 136)	64	72	36	52	32	16

Velocidade de avanço

Foram escolhidas duas velocidades distintas, 1200 e 12000 mm/min, identificadas nos testes como v20 e v200 (mm/s), respetivamente. A velocidade mais baixa pode considerar-se representativa da ordem de grandeza de velocidades de avanço na maquinagem de alumínio [42]. A escolha de uma velocidade dez vezes superior à recomendada relaciona-se com um dos objetivos deste trabalho: a implicação da velocidade de avanço na descrição de trajetórias.

5.2 Caso de estudo 1: Análise do movimento linear com 3 pontos

Objetivo: Analisar os percursos e os perfis de velocidade em movimentos lineares com 3 pontos, variando a velocidade de avanço, *zone*, G05, G51 e usando as funções de interpolação linear disponíveis (G01 e MoveL). Em relação à aceleração mantiveram-se os valores definidos por defeito para o CNC e RI. Os valores de velocidades e tempos de execução relativos ao robô foram obtidos da simulação (*Signal Analyzer*). Os valores do CNC são recolhidos ao longo da realização da trajetória, na mesa XY (*Oscilloscope*).

Percurso em linha

Foi criada uma linha constituída por três pontos no plano horizontal, sendo P2 o ponto intermédio (Figura 5.3) de comprimento total igual a 280 mm. A execução desta trajetória em linha envolveu apenas um eixo de movimento no CNC.

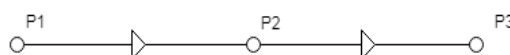


Figura 5.3 - Percurso implementado - linha.

Foram realizadas simulações para ambos os sistemas, tendo os respetivos tempos de execução sido anotados na Tabela 5.3 para efeitos de comparação.

Tabela 5.3 - Tempos de execução do percurso em linha

Velocidade	Tempos [s]						
	RI			CNC			
	<i>fine</i>	z1	z10	G01	G05	G51 E1.0	G51 E10.0
20 mm/s	14,38	14,02	14,02	14,23	13,90	13,93	13,93
200 mm/s	1,63	1,54	1,54	2,33	1,66	1,69	1,69

Com na base nos resultados obtidos, verifica-se que:

- À velocidade mais baixa (20 mm/s), os tempos de execução são muito próximos nas duas máquinas, em qualquer dos parâmetros considerados.

- À velocidade mais elevada, o tempo de execução do CNC é superior ao do robô (aproximadamente 43%), para G01/*fine*. Este comportamento permite tempos de execução inferiores, pois o robô tem uma aceleração superior.
- Quando são usados parâmetros que não obrigam à paragem no ponto intermédio, os tempos são praticamente constantes em cada uma das máquinas.
- A função *look-ahead*, comparativamente com a função G05, apresenta tempos apenas marginalmente superiores.

Trajetórias executadas com o RI

De seguida, apresentam-se na Figura 5.4 e na Figura 5.5, parte das trajetórias executadas, centradas em torno do ponto P2.

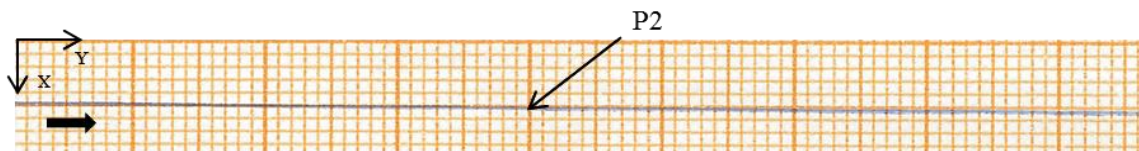


Figura 5.4 – Representação parcial dos 3 percursos em linha realizados pelo robô para 20 mm/s (*fine*, *z1* e *z10*).

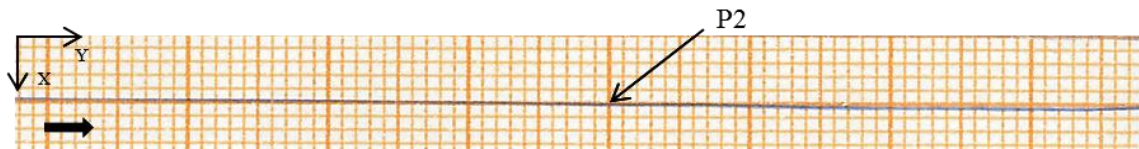


Figura 5.5 – Representação parcial dos 3 percursos em linha realizados pelo robô para 200 mm/s (*fine*, *z1* e *z10*).

Dada a grande semelhança entre ambos os cenários, as conclusões retiradas são idênticas:

- O parâmetro *zone* não afeta o comportamento do robô na descrição da trajetória em torno do ponto P2, visto os 3 percursos serem coincidentes;
- A velocidade programada não afeta significativamente o comportamento do robô na descrição do percurso em torno do ponto P2.

Seguidamente, foram representados os perfis de velocidade para as duas velocidades propostas (Figura 5.6 e Figura 5.7), com dois valores de *zone* (*z1* e *z10*), concluindo-se que o parâmetro *zone* não condiciona a velocidade programada para a trajetória, uma vez que se mantém constante em torno do ponto intermédio P2.

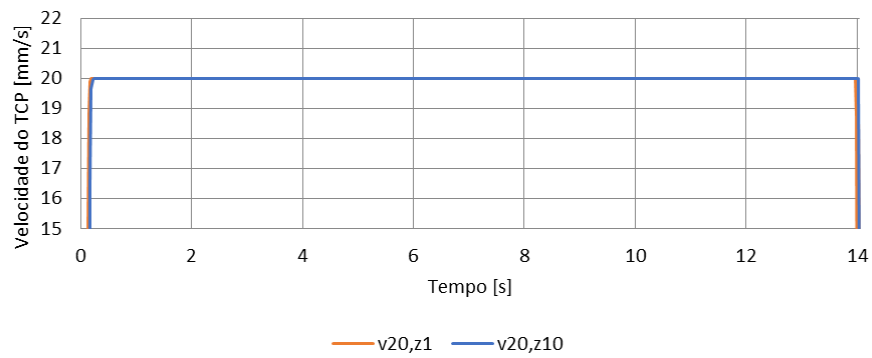


Figura 5.6 – Excerto dos perfis de velocidade do TCP durante a execução de uma linha para 20 mm/s ($z1$ e $z10$).

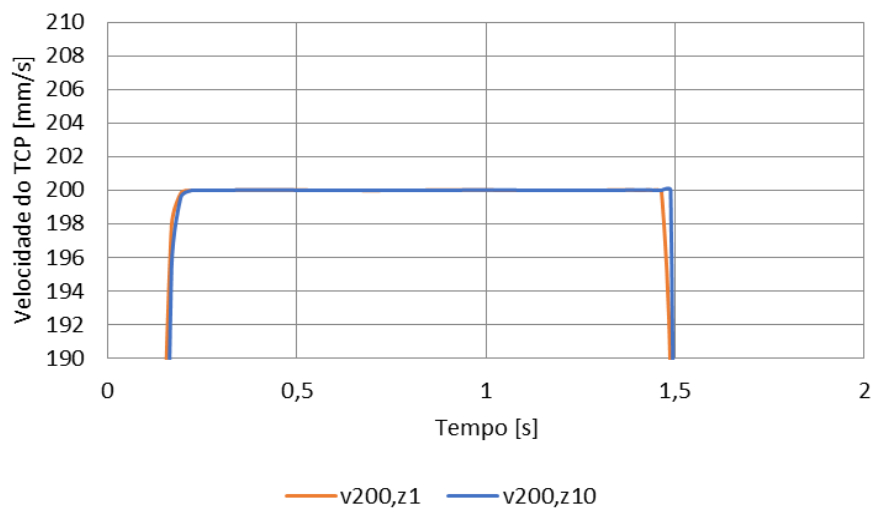


Figura 5.7 – Excerto dos perfis de velocidade do TCP durante a execução de uma linha para 200 mm/s ($z1$ e $z10$).

Trajetórias executadas com o CNC

A Figura 5.8 e a Figura 5.9 exibem os percursos em linha, descrita pelo CNC em torno do ponto intermédio P2.

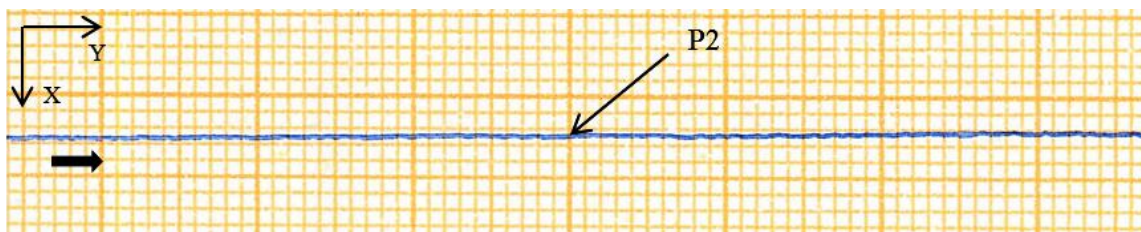


Figura 5.8 – Representação parcial dos 4 percursos obtidos usando as funções G01, G05 e G51 (E1.0 e E10.0) para 20 mm/s.

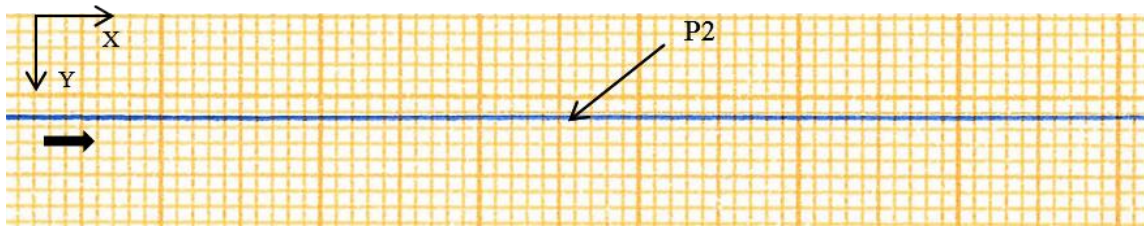


Figura 5.9 – Representação parcial dos 4 percursos obtidos usando as funções G01, G05 e G51 (E1.0 e E10.0) para 200 mm/s.

Verifica-se que nas velocidades testadas, não existe qualquer perturbação ao nível dos percursos obtidos, associada à existência de um ponto intermédio. Pode-se concluir, por isso, que a existência do ponto P2 é irrelevante face ao uso da função G05.

Na Figura 5.10 e na Figura 5.11 estão representados os perfis da velocidade de avanço, relativos ao uso da função de arredondamento de aresta (G05).

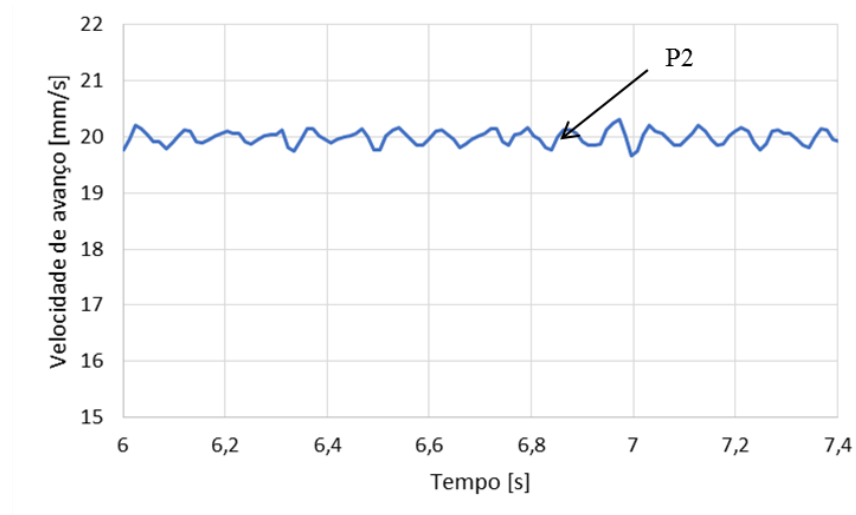


Figura 5.10 – Excerto do perfil da velocidade de avanço na execução de um percurso em linha para 20 mm/s usando a função arredondamento de aresta (G05).

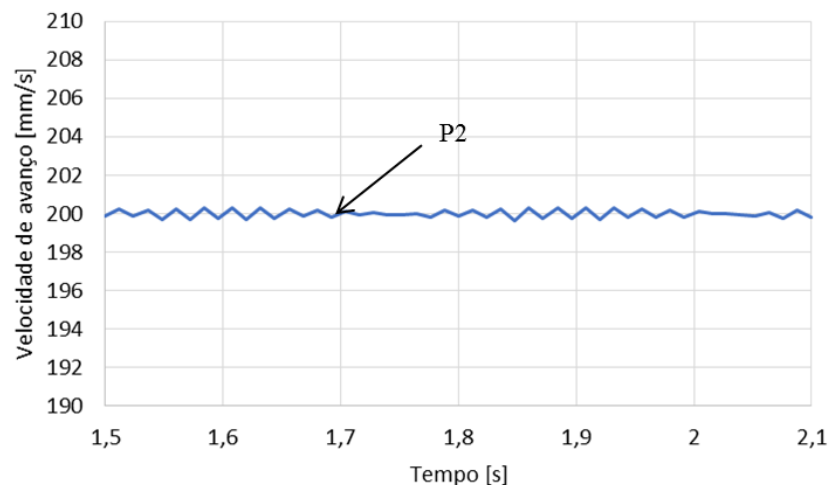


Figura 5.11 – Excerto do perfil da velocidade de avanço na execução de um percurso em linha para 200 mm/s usando a função arredondamento de aresta (G05).

Pode-se depreender da análise da figura anterior que, à semelhança do que acontece com os percursos obtidos, não existe qualquer perturbação da velocidade de avanço associada à existência de P2. Assim, pode-se inferir que a inclusão de P2 não tem influência na descrição do percurso, quando a função G05 está ativa, para as duas velocidades testadas.

Na Figura 5.12 e na Figura 5.13 estão expostos os perfis da velocidade de avanço relativos ao uso da função *look-ahead* (G51), configurada com E0.1 e E10.0. De forma análoga aos casos anteriores, pode-se verificar que a velocidade se mantém aproximadamente constante, independentemente da velocidade programada e do erro de contorno especificado.

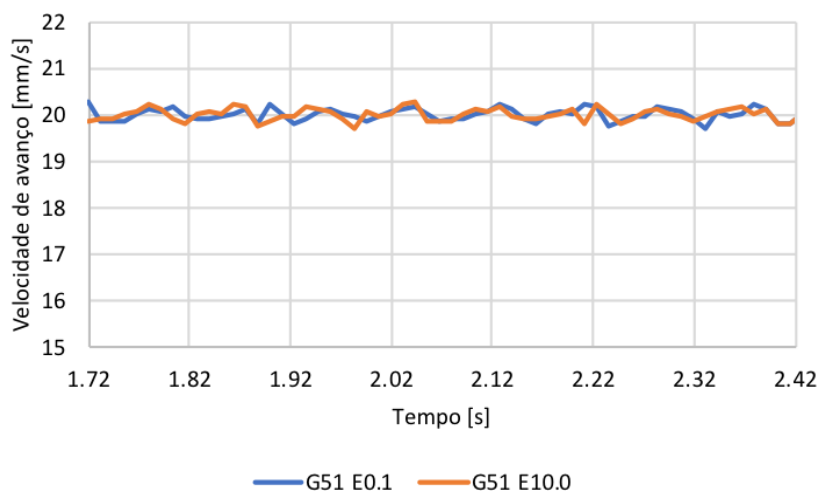


Figura 5.12 – Excerto dos perfis da velocidade de avanço na execução de percursos em linha para 20 mm/s e usando a função *look-ahead* (G51).

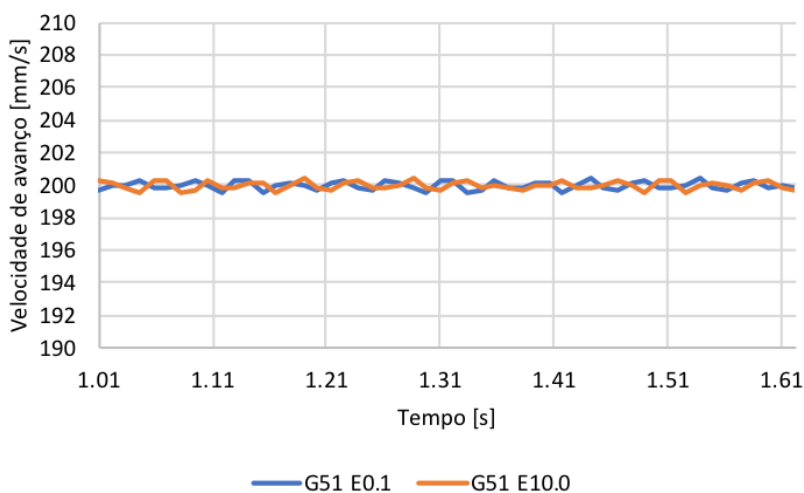


Figura 5.13 – Excerto dos perfis da velocidade de avanço na execução de percursos em linha para 200 mm/s e usando a função *look-ahead* (G51).

Síntese dos resultados: movimento em linha

Verifica-se que, introduzindo um ponto intermédio colinear, com o objetivo de condicionar a execução de uma trajetória linear usando as funcionalidades testadas, não tem significado a exceto as que obrigam a parar nesse ponto.

Percurso em canto

Foi definido um percurso em canto de 180x140 mm (Figura 5.14). No CNC, o percurso evidenciado é paralelo aos eixos coordenados (XY). Os parâmetros estudados e testes realizados correspondem aos definidos para o percurso em linha.

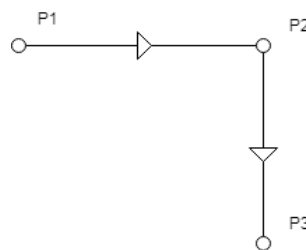


Figura 5.14 – Percurso implementado – canto.

Foram anotados os tempos de execução nos dois sistemas na Tabela 5.4.

Tabela 5.4 – Tempos de execução do percurso em canto

Velocidade	Tempos [s]						
	RI			CNC			
	<i>fine</i>	z1	z10	G01	G05	G51 E1.0	G51 E10.0
20 mm/s	16,37	15,94	15,79	16,47	15,94	15,94	15,92
200 mm/s	1,82	1,70	1,66	2,57	1,88	2,26	2,18

Pode-se concluir que as diferenças mais significativas se encontram no caso do CNC, para a velocidade mais alta (200 mm/s), onde existe uma disparidade nos tempos de execução obtidos.

Trajétórias executadas com RI

Para velocidades de 20 e 200 mm/s e parâmetros *zone fine*, *z1*, *z5* e *z10* obtiveram-se os percursos representados na Figura 5.15.

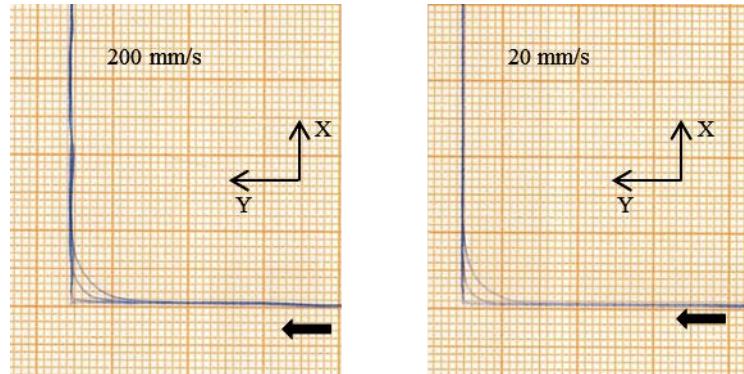


Figura 5.15 – Representação parcial das posições do TCP para oito percursos (*fine*, *z1*, *z5* e *z10*).

Através da observação da figura anterior, pode-se verificar que:

- O parâmetro *zonedata* diferente de *fine* provocou alterações do percurso que se podem associar à descrição de um segmento circular de raio igual ao valor de *zonedata*, não sendo influenciadas pela velocidade.

Na Figura 5.16 e na Figura 5.17 estão representados os perfis de velocidades do TCP, durante a execução do canto programado.

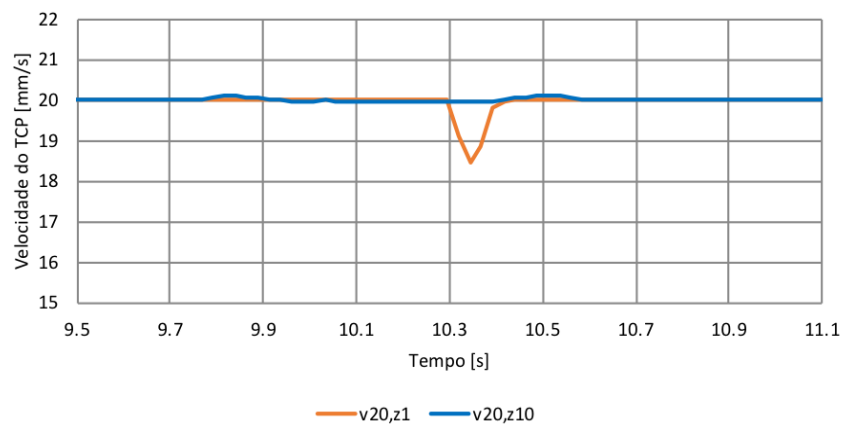


Figura 5.16 – Excerto dos perfis de velocidade do TCP durante a execução de um percurso em canto para 20 mm/s (*z1* e *z10*).

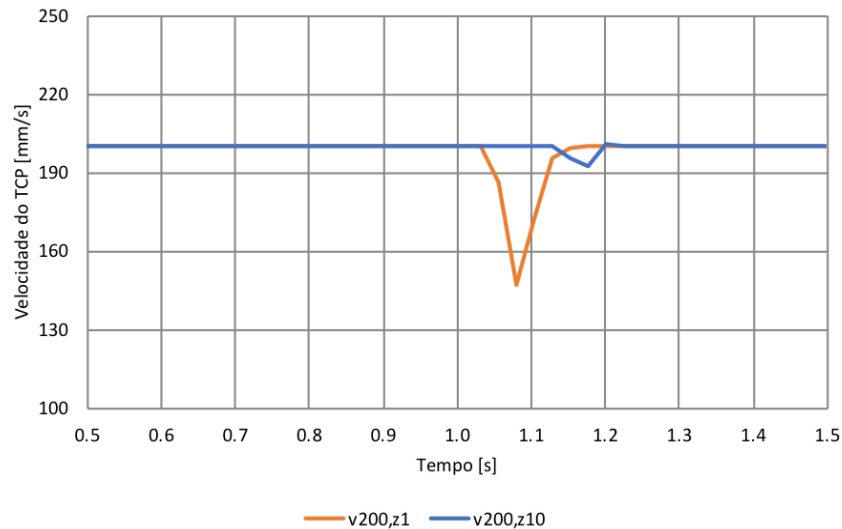


Figura 5.17 – Excerto dos perfis de velocidade do TCP durante a execução de um percurso em canto para 200 mm/s ($z1$ e $z10$).

De acordo com a análise dos perfis de velocidade anteriores, pode-se induzir que, para 20 mm/s e $z1$, houve uma redução da velocidade face ao seu valor programado de 8% e para o caso de $z10$ a atenuação de velocidade foi impercetível. Para 200 mm/s e recorrendo aos parâmetros de $zone$, $z1$ e $z10$, houve uma redução de velocidade de aproximadamente 27% e 4% face à velocidade programada, respetivamente. Verifica-se que para o mesmo parâmetro $zone$, a redução de velocidade é tanto maior, quanto maior for a velocidade programada.

Trajetórias executadas com o CNC

Para velocidades de 20 e 200 mm/s, usando as funções arredondamento de aresta (G05) e *look-ahead* (G51), obtiveram-se os percursos representados na Figura 5.18, Figura 5.22 e na Figura 5.23.

À semelhança do robô, quando não é especificado o tipo de aresta a executar no CNC, este acaba obrigatoriamente por parar na passagem pelo canto, atingindo, por isso, velocidade nula. Na Figura 5.18 estão representadas as trajetórias executadas pelo CNC, usando a função de movimento linear (G01) e a função de arredondamento de aresta (G05).

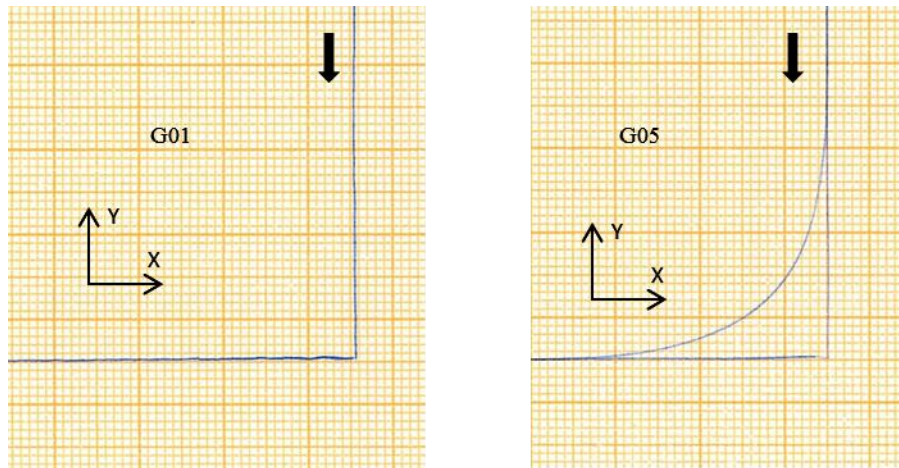


Figura 5.18 – Representação parcial dos 4 percursos obtidos recorrendo ao uso das funções G01 e G05 (20 e 200 mm/s).

Analisando simultaneamente ambas, pode-se concluir que:

- O percurso segue um percurso linear sem a ocorrência de variações posicionais significativas, para os casos em que apenas foi usada a função de movimento linear (G01), sendo o percurso coincidente para as duas velocidades programadas, no entanto, notam-se algumas perturbações no início do movimento em relação ao eixo Y;
- O percurso obtido afasta-se significativamente do programado, quando a função de arredondamento de aresta está ativa, para uma velocidade de 200 mm/s.

Para avaliar melhor a evolução do desvio do percurso em função da velocidade, quando se usa a função G05, efetuaram-se mais alguns testes com o CNC a diferentes velocidades, tendo os resultados sido apresentados no gráfico da Figura 5.19.

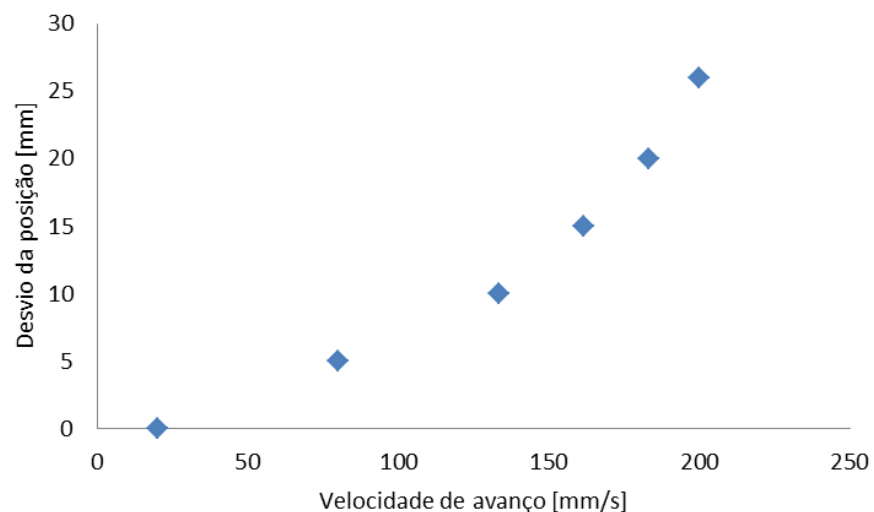


Figura 5.19 – Desvio da posição em função da velocidade de avanço, para o percurso em canto.

Na Figura 5.20 e na Figura 5.21 estão representados os perfis de velocidade de avanço obtidos a partir da função *Oscilloscope*.

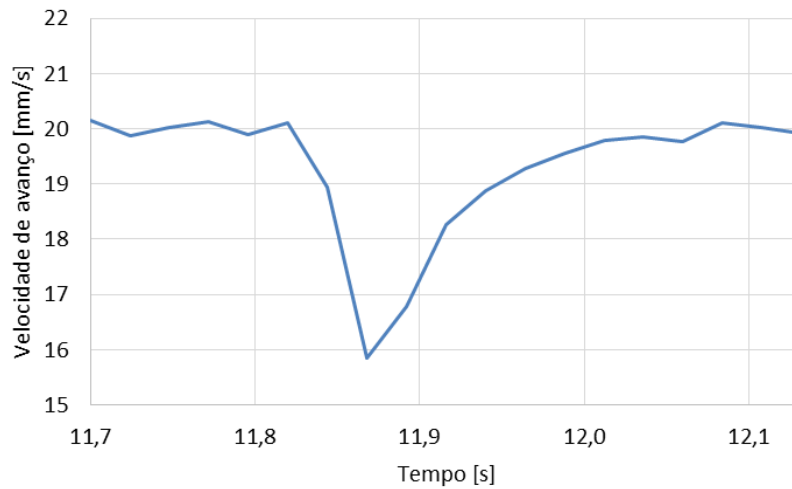


Figura 5.20 – Excerto do perfil da velocidade de avanço na execução de um percurso em canto para 20 mm/s, usando a função de arredondamento de aresta (G05).

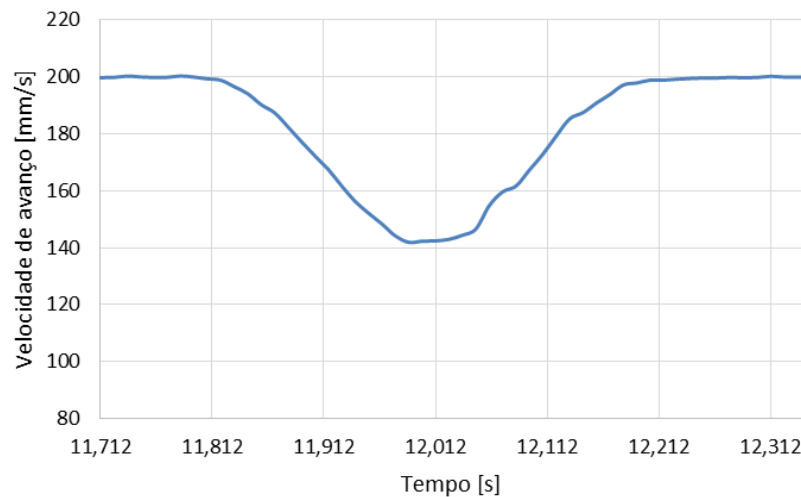


Figura 5.21 - Excerto do perfil da velocidade de avanço na execução de um percurso em canto para 200 mm/s, usando a função de arredondamento de aresta (G05).

Da análise conjunta efetuada, apura-se que, para 20 mm/s, durante a mudança de direção, houve uma diminuição da velocidade na ordem de aproximadamente 4 mm/s, o que acarreta uma quebra de velocidade de 21%. Para 200 mm/s, a velocidade de avanço teve um decréscimo de 29% face ao seu valor programado.

Atendendo agora ao uso da função *look-ahead*, foram realizados dez testes para as velocidades 20 e 200 mm/s. Os percursos obtidos experimentalmente encontram-se representadas na Figura 5.22 e na Figura 5.23.

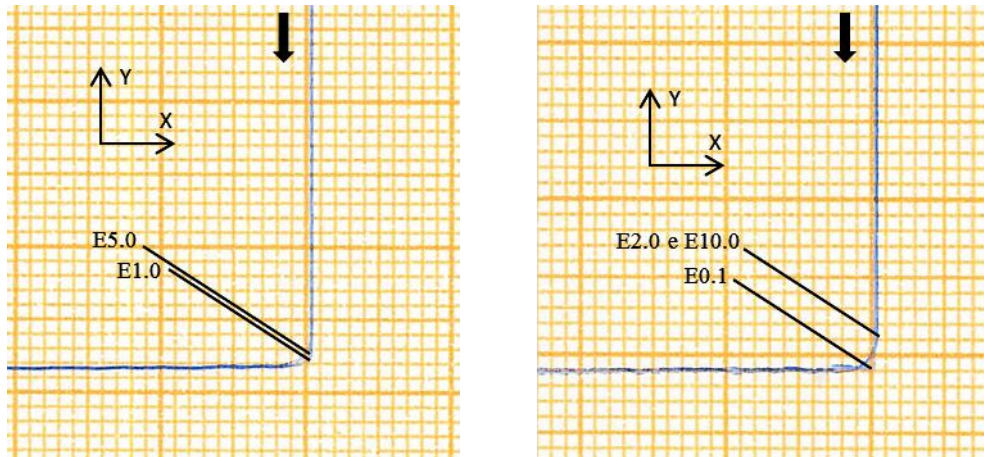


Figura 5.22 – Representação parcial dos percursos obtidos para 20 mm/s recorrendo à função *look-ahead* (G51).

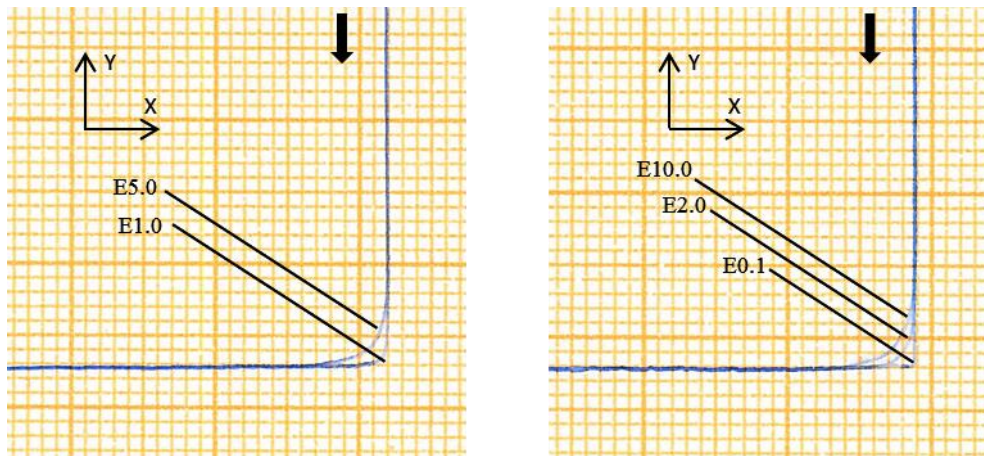


Figura 5.23 – Representação parcial dos percursos obtidos para 200 mm/s recorrendo à função *look-ahead* (G51).

Analisando as figuras anteriores, verifica-se que:

- Existe um arredondamento do percurso mais pronunciado à velocidade mais elevada, no entanto, não é igual ao erro de contorno especificado;
- É clara a influência da velocidade de avanço e do erro de contorno (E).

Na Figura 5.24 e na Figura 5.25 estão representados os perfis da velocidade de avanço para os dois casos extremos de erro de contorno.

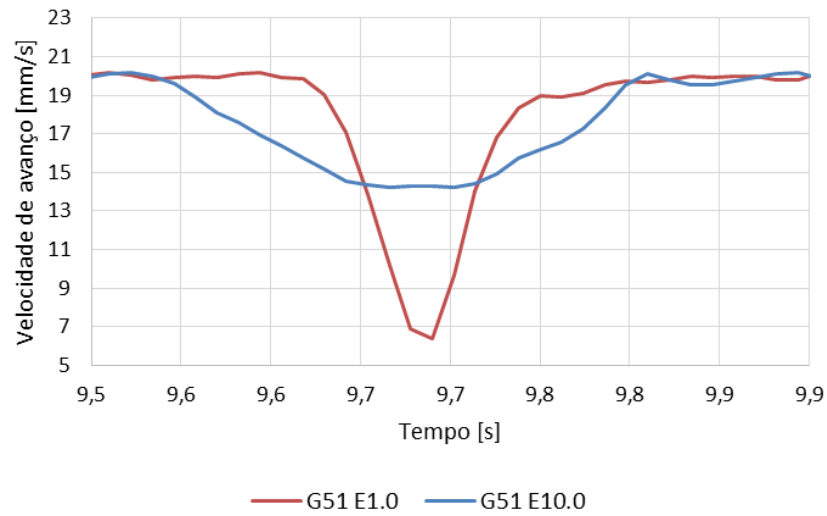


Figura 5.24 – Excerto dos perfis da velocidade de avanço na execução de percursos em canto para 20 mm/s, usando a função *look-ahead* (G51).

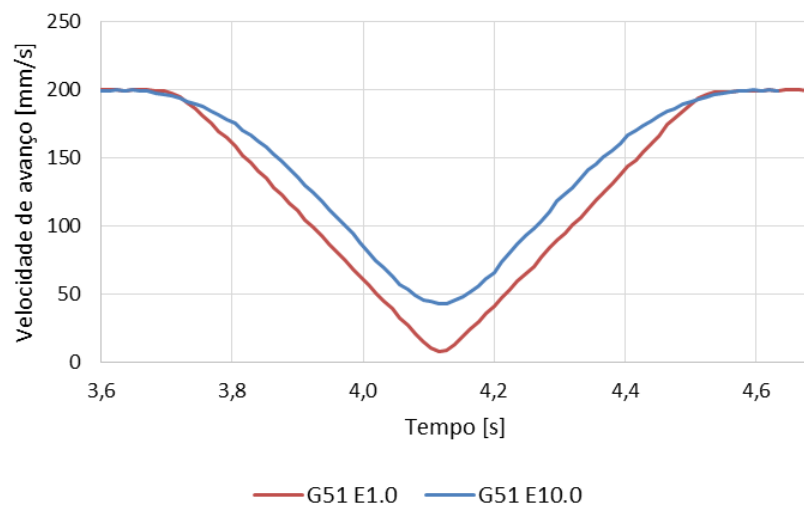


Figura 5.25 – Excerto dos perfis da velocidade de avanço na execução de percursos em canto para 200 mm/s, usando a função *look-ahead* (G51).

Analisando os resultados obtidos, pode-se inferir que:

- Para 20 mm/s, a velocidade de avanço teve um decréscimo de 78% e 29%, para erros de contorno de 0.1 e 10 mm, respetivamente;
- Para 200 mm/s, a velocidade de avanço teve um decréscimo de 96% e 78%, para erros de contorno iguais aos citados anteriormente.

Síntese dos resultados: movimento em canto

Em relação ao tempo de execução dos percursos, o robô apresenta em geral tempos menores para as duas velocidades programadas, o que, tal como no caso anterior, leva a inferir que a aceleração potenciada pelo RI é superior à do CNC.

Relativamente aos perfis de velocidade, de um modo geral estão associadas maiores reduções de velocidades, a velocidades programadas superiores, nos dois sistemas. Contrariamente ao CNC, o robô conseguiu responder melhor às mudanças de direção através de menores reduções de velocidade e, simultaneamente, menores desvios nos percursos. A título de exemplo, para 200 mm/s, usando a função arredondamento de aresta, o CNC conseguiu ter um desvio correspondente a um raio de concordância igual a 26 mm, com uma diminuição da velocidade de 29%, enquanto que o robô viu o seu percurso alterado com um desvio de 1 mm e uma redução de velocidade de 27%. Com o uso da função *look-ahead*, os perfis de velocidade revelaram ser mais “pontagudos”, com uma maior redução de velocidade, o que se traduziu num menor desvio da trajetória.

Em relação ao percurso, este é apenas alterado na zona circundante do canto, sendo que, no caso do robô, este é apenas influenciado pelo parâmetro *zone*, independentemente da velocidade programada. Enquanto no caso do CNC, este depende das funções usadas - arredondamento de aresta (G05) e *look-ahead* (G51) – e da velocidade de avanço programada, correspondendo uma maior diferença entre os percursos real e teórico, a uma maior velocidade de avanço.

5.3 Caso de estudo 2: Análise do movimento em U

Objetivo: Analisar os percursos obtidos com movimentos lineares com 4 pontos e movimentos lineares/circulares com 4/5 pontos, variando a velocidade de avanço, *zone*, arredondamento de aresta (G05) e *look-ahead* (G51), usando as funções de interpolação linear e circular, disponíveis nos dois sistemas (G01/G02/G03 e MoveL/MoveC). Os parâmetros estudados e testes realizados correspondem a todos os valores enunciados na escolha de parâmetros. Em relação à avaliação de resultados dos testes realizados, em todos os restantes casos (2 a 4), apenas são considerados dados de posição e tempos de execução.

Na Figura 5.26 estão representados os percursos definidos para o caso de estudo 2. Onde a distância de P1 a P2 e P2 a P3 é 140 mm e 5 mm, respetivamente.

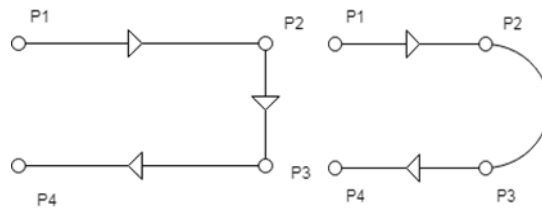


Figura 5.26 – Percurso em U de base reta (à esquerda) de base circular (à direita).

Percurso em U de base reta

Apresentam-se os tempos de percurso dos dois sistemas na Tabela 5.5.

Tabela 5.5 – Tempos de execução do percurso em U de base reta

Velocidade	Tempos [s]						
	RI			CNC			
	<i>fine</i>	<i>z1</i>	<i>z10</i>	G01	G05	G51 E1.0	G51 E10.0
20 mm/s	14,40	14,26	14,21	14,64	14,15	14,27	14,18
200 mm/s	1,70	1,61	1,56	2,64	2,07	2,24	2,11

Analisando a tabela anterior verifica-se novamente a predominância do robô na obtenção de tempos de execução inferiores.

Trajetórias executadas com o RI

Para uma largura de base de 5 mm, velocidades de 20 e 200 mm/s, e para diferentes valores de *zone*, foram obtidos os resultados representados na Figura 5.27 e na Figura 5.28.

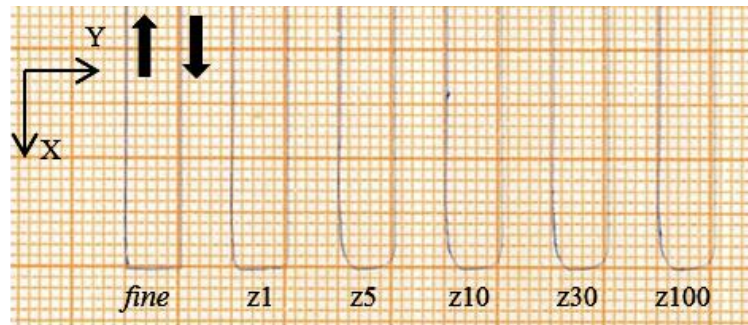


Figura 5.27 – Representação parcial dos percursos em U de base reta para 20 mm/s (*fine* e *z1* a *z100*).

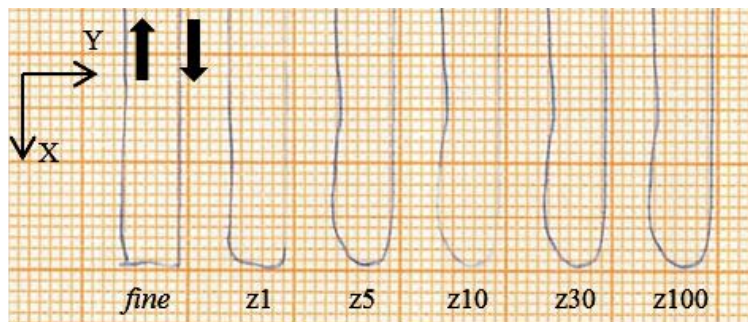


Figura 5.28 – Representação parcial dos percursos em U de base reta para 200 mm/s (*fine* e *z1* a *z100*).

Analisando atentamente os resultados recolhidos, conclui-se que:

- Para 20 mm/s e até *z1*, os percursos são pouco afetados, tal como esperado. Para parâmetros de *zone* superiores a *z5*, todas as trajetórias são aparentemente coincidentes;
- Para 200 mm/s são notórias maiores alterações de posição face ao percurso programado.

Trajetórias executadas com o CNC

O mesmo caso foi implementado no CNC, dando origem a catorze percursos diferentes como evidenciado na Figura 5.29 e na Figura 5.30.

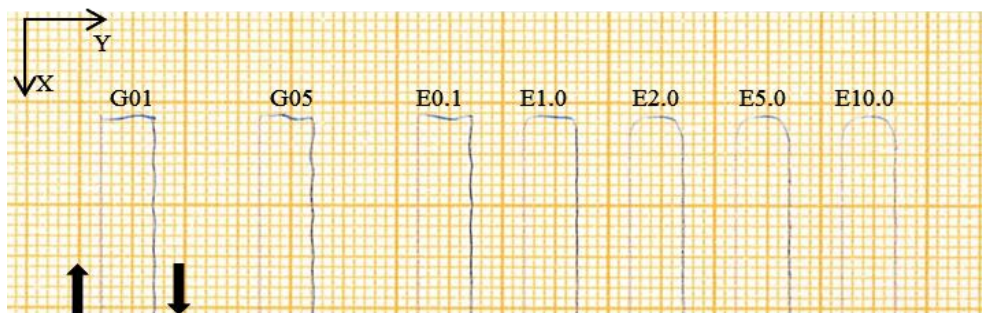


Figura 5.29 – Representação parcial dos percursos em U de base reta para 20 mm/s (G01, G05 e G51).

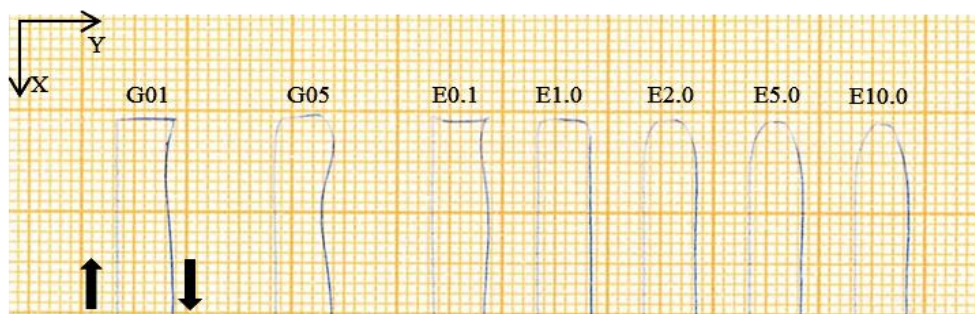


Figura 5.30 – Representação parcial dos percursos em U de base reta para 200 mm/s (G01, G05 e G51).

Comparando os resultados recolhidos, pode-se inferir que:

- Para 20 mm/s, G01 e G05, o percurso programado é efetuado de forma semelhante. Isto pode ser explicado pela proximidade do ponto P3 relativamente a P2, atenuando o resultado da utilização de G05. Em relação à função G51 é notória a influência dos parâmetros E nos desvios de posição, sendo sempre atingido o ponto intermédio do segmento P2P3;
- Para 200 mm/s, os desvios de posição são amplificados devido ao aumento de velocidade.

Síntese dos resultados: movimento em U de base reta

Genericamente, é novamente demonstrado que maiores variações dos percursos estão associadas a maiores velocidades, maiores erros de contorno e maiores parâmetros *zone* especificados. Excluindo o arredondamento de aresta, há ainda a ondulação evidenciada no movimento de retorno, ou seja, depois da execução da base (segmento P2P3). Esta é originada, em parte, pela folga existente na caneta usada para executar as várias trajetórias. Esta ocorrência apresenta uma variação mais acentuada no CNC, para a qual também contribui a baixa rigidez da mesa de suporte e eventuais folgas no sistema de transmissão dos eixos da mesa XY.

De um modo geral, os dois sistemas conseguem atingir a cota programada, sendo que a partir de um certo valor de *zone* ou erro de contorno (E), os percursos são praticamente coincidentes.

Percurso em U de base circular

Dado que todos os testes anteriores envolvem uma base reta e que a variação da sua parametrização conduz ao arredondamento dos seus cantos, é agora tempo de avaliar os efeitos da parametrização na descrição de um movimento em “U” com base circular. Como foi visto anteriormente na programação de um movimento circular, usando o robô, é necessário especificar um ponto intermédio (Figura 5.31). No CNC isto não acontece, sendo que apenas é necessário alterar a função G01 por G02 com especificação das coordenadas do centro da circunferência relativamente a P2. Na Figura 5.32 é representado o percurso e respetivo programa implementados no CNC. Os parâmetros estudados e testes realizados correspondem aos definidos para o percurso em U de base reta.

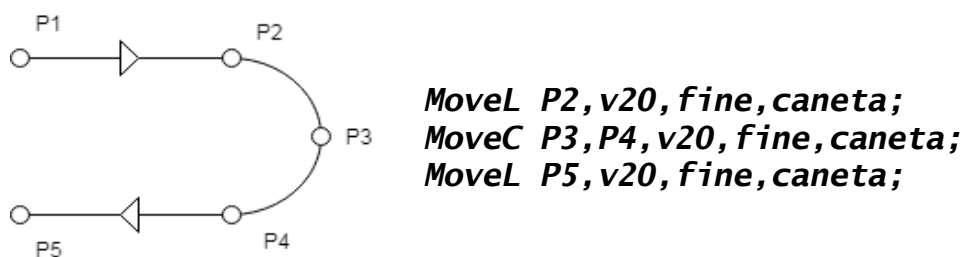


Figura 5.31 – Percurso e respetivo programa, implementados no RI – movimento em U de base circular.

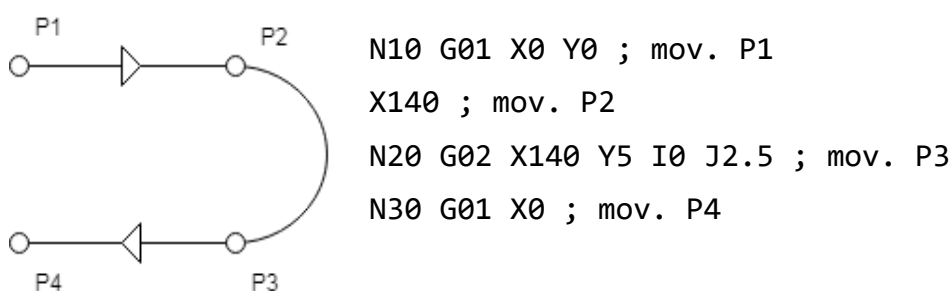


Figura 5.32 – Percurso e respetivo programa, implementados no CNC – movimento em U de base circular.

Os tempos de percurso estão espelhados na Tabela 5.6, onde se podem retirar as mesmas conclusões referentes ao movimento em U de base reta.

Tabela 5.6 – Tempos de execução do percurso em U de base circular

Velocidade	Tempos [s]						
	RI			CNC			
	<i>fine</i>	z1	z10	G01/G02	G05	G51 E1.0	G51 E10.0
20 mm/s	14,54	14,45	14,40	14,76	14,30	14,32	14,32
200 mm/s	1,75	1,63	1,61	2,71	2,21	2,18	2,18

Trajétórias executadas com o RI

Foram realizados dez testes envolvendo diferentes parâmetros *zone* e duas velocidades. Os resultados da posição estão evidenciados na Figura 5.33 e na Figura 5.34.

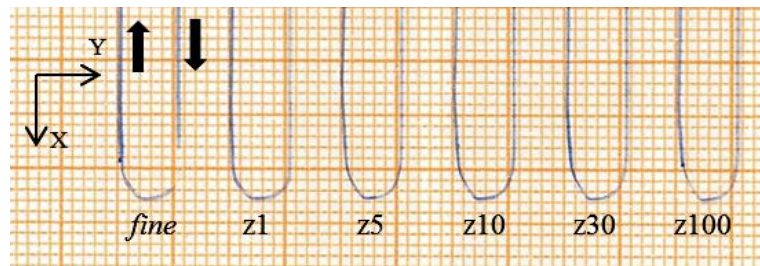


Figura 5.33 – Representação parcial dos percursos em U de base circular para 20 mm/s (*fine* e *z1* a *z100*).

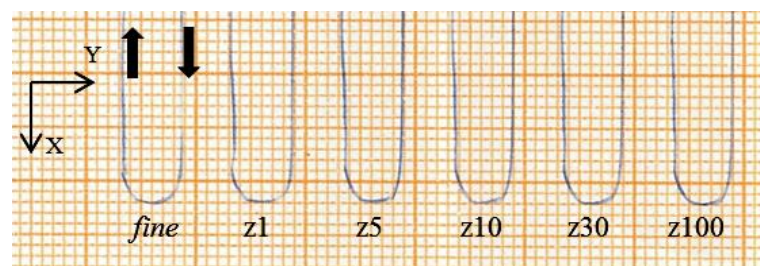


Figura 5.34 – Representação parcial dos percursos em U de base circular para 200 mm/s (*fine* e *z1* a *z100*).

Da análise da representação gráfica anterior, conclui-se que:

- O aumento do parâmetro *zone* não conduz a desvios significativos entre os vários percursos realizados;
- De um modo geral, o percurso foi ligeiramente mais distorcido para a velocidade de 200 mm/s.

Trajétórias executadas com o CNC

Foram realizados catorze testes a fim de comprovar o comportamento do CNC, ao descrever um movimento em U com base circular, tendo os resultados obtidos sido representados na Figura 5.35 e na Figura 5.36.

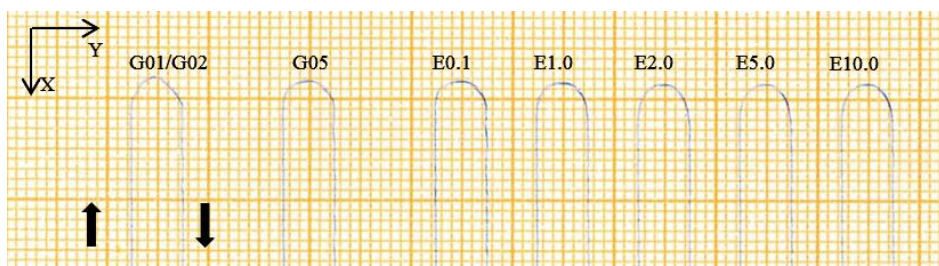


Figura 5.35 – Representação parcial dos percursos em U de base circular para 20 mm/s (G01/G02, G05 e G51).

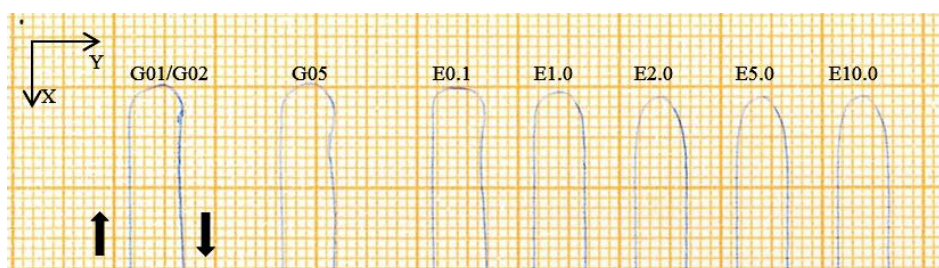


Figura 5.36 – Representação parcial dos percursos em U de base circular para 200 mm/s (G01/G02, G05 e G51).

Fazendo a análise comparativa de ambas, conclui-se que:

- Para 20 mm/s, os percursos são realizados de forma similar, aproximando-se do percurso programado, para os vários testes;
- Para 200 mm/s, os percursos executados apresentam um desvio maior relativamente ao percurso programado. Esse desvio acentua-se para valores de E maiores, sendo também evidente que não se alcança o valor limite.

Síntese de resultados: movimento em U de base circular

Relativamente ao robô verifica-se que o parâmetro *zone* teve uma menor implicação nos movimentos circulares (U com base circular), relativamente aos movimentos lineares (U com base reta), facto que é justificado pelo parâmetro *zone*.

No que concerne ao CNC, a utilização da função G05 (arredondamento de aresta), tem impacto no tempo de execução do percurso, apesar deste já ser circular, uma vez que evita a paragem na transição entre segmentos (linear/circular).

5.4 Caso de estudo 3: Análise do movimento circular

Objetivo: Analisar os percursos obtidos com movimentos circulares definidos de modo distinto, variando a velocidade de avanço, *zone*, *AccSet*, arredondamento de aresta (G05) e *look-ahead* (G51), usando as funções de interpolação circular disponíveis nos dois sistemas (G02/G03 e MoveC).

Foi criado um percurso envolvendo várias circunferências de raio 10 mm (Figura 5.37). Tal como foi explicado anteriormente (Capítulo 4.4), no robô, uma circunferência é definida por 4 pontos correspondendo a 2 blocos de programação MoveC (Figura 5.37), enquanto para o caso do CNC foi definido usando as coordenadas de um ponto (P1) e do centro da circunferência em relação a esse ponto.

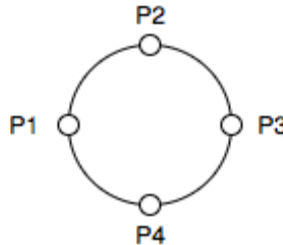


Figura 5.37 – Percurso programado para o caso do robô.

De seguida, apresentam-se na Tabela 5.7 os tempos de execução dos vários testes realizados.

Tabela 5.7 – Tempos de execução de realização de uma circunferência

Velocidade	Tempos [s]						
	RI			CNC			
	<i>fine</i>	z1	z10	G02	G05	G51 E1.0	G51 E10.0
20 mm/s	3,29	3,14	3,14	3,27	3,04	3,06	3,06
200 mm/s	0,53	0,43	0,43	1,12	0,88	0,92	0,92

A partir da análise da Tabela 5.7 pode-se verificar que:

- O robô tem acelerações mais elevadas, o que pode ser constatado não só a velocidades elevadas, mas também à velocidade mais baixa. A esta velocidade (20 mm/s), o tempo de percurso do robô (*fine*) é só marginalmente superior ao do CNC (G02), apesar de parar nos pontos intermédios, enquanto o CNC não tem pontos intermédios;
- Em relação às funções G05 e G51, estas apresentam tempos semelhantes, comparativamente com os obtidos com a função G02.

Trajetórias executadas com o RI

Foram realizados seis ensaios envolvendo diferentes parâmetros *zone* a duas velocidades diferentes. Neste âmbito, os resultados da variação da posição estão evidenciados na Figura 5.38 e na Figura 5.39.

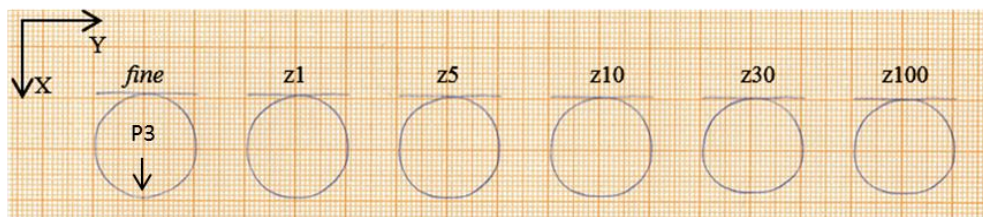


Figura 5.38 – Percursos circulares para 20 mm/s (*fine* e *z1* a *z100*).

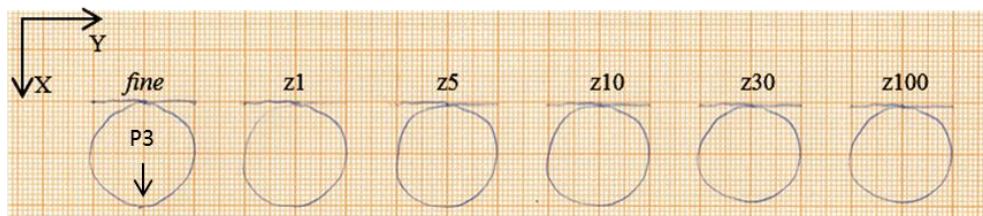


Figura 5.39 – Percursos circulares para 200 mm/s (*fine* e *z1* a *z100*).

Tendo em conta os resultados anteriores, pode-se inferir que:

- Para as duas velocidades programadas, os percursos executados afastam-se dos programados na zona envolvente do ponto P3, devido ao parâmetro *zone*;
- Os desvios de forma nos percursos executados são mais visíveis à velocidade mais elevada.

De maneira a estudar a influência do parâmetro *AccSet* foram repetidos os casos de estudo relativos ao movimento circular para uma velocidade de 200 mm/s, onde se verificou que a grande aceleração deteriorava a descrição das trajetórias. Neste âmbito, foi alterado o parâmetro *AccSet 100,100* para *AccSet 10,10*, resultando nos percursos da Figura 5.40. É visível que a forma do percurso descrito se aproxima mais do programado.

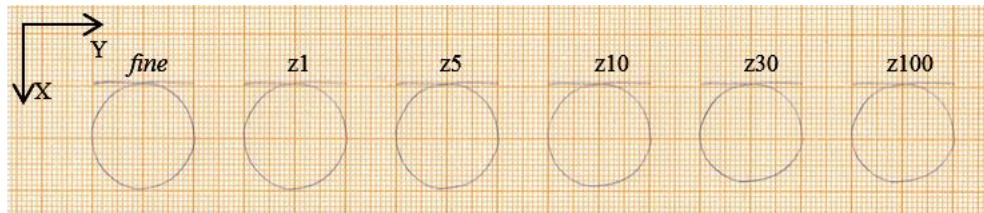


Figura 5.40 – Percursos circulares para 200 mm/s (*fine* e *z1* a *z100*) com um parâmetro *AccSet 10,10*.

Trajétórias executadas com o CNC

Foram realizados catorze testes a fim de investigar o comportamento do CNC ao descrever um movimento em circunferência, tendo os resultados sido representados na Figura 5.41 e na Figura 5.42.

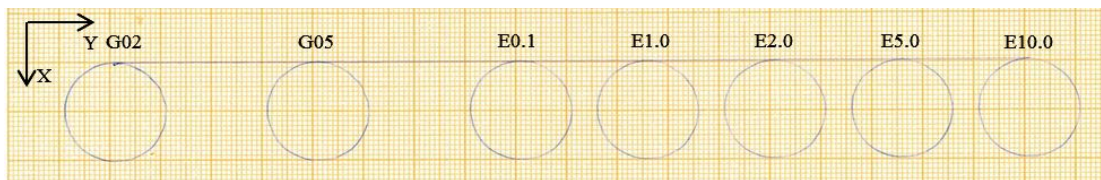


Figura 5.41 – Percursos circulares para 20 mm/s (G02, G05 e G51).

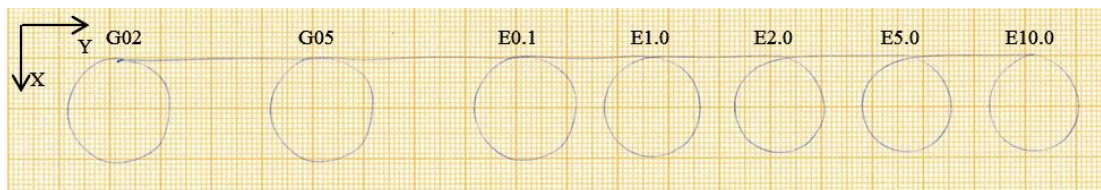


Figura 5.42 – Percursos circulares para 200 mm/s (G02, G05 e G51).

Fazendo um estudo comparativo entre as trajetórias anteriores, depreende-se que:

- Para 20 mm/s, os percursos conseguidos têm uma elevada semelhança entre eles, conseguindo este descrever praticamente uma circunferência perfeita;
- Para 200 mm/s, os percursos obtidos apresentam alguma deformação devido à parametrização. No caso *fine* e G05, o percurso já se afasta mais do percurso

programado. Já para o caso *look-ahead* existe um desvio que é agravado pelo aumento do erro de contorno.

Síntese de resultados: movimento circular

O facto de no CNC não ser necessário especificar pontos intermédios para definir circunferências, permite uma execução dos percursos mais próxima do programado.

5.5 Caso de estudo 4: Análise de estratégias de maquinagem









Objetivo: Examinar o desempenho das diferentes estratégias de maquinagem para a execução de caixas.

Com vista à concretização deste objetivo, recorre-se ao Mastercam/Robotmaster para definir um quadrado de lado 150 mm e gerar programas de maquinagem com base em 8 das estratégias de maquinagem disponibilizadas. Nesse sentido, pretende-se avaliar os princípios e funcionalidades da programação de percursos de maquinagem quando se geram os programas para o robô ou para o CNC. Foram aplicadas as mesmas condições de maquinagem para as oito alternativas, tendo sido programada uma velocidade de avanço de 20 mm/s e usados os parâmetros *z1* e G05, para o robô e CN, respetivamente. É de realçar que estes valores foram escolhidos para minimizar a redução de velocidade e desvio em relação ao percurso programado. As estratégias de maquinagem de caixas agrupam-se em oito categorias:

1. *Zig-zag*
2. Espiral com sobreposição constante
3. Espiral paralela
4. Espiral paralela com cantos retocados
5. Espiral de *Morph*
6. Alta velocidade
7. Sentido único
8. Espiral real

A implementação nos dois sistemas foi realizada a partir dos programas gerados no *software*. Em relação ao robô já estava contemplada a especificação do parâmetro *zone* (zI), enquanto que em relação ao CNC foi necessário editar o código para incorporar a função G05. Os resultados, apresentados na Tabela 5.8, incluem o número de linhas do programa executado em cada sistema. A distância de maquinagem diz respeito apenas à distância percorrida em operação de corte e foi obtida a partir do *software*:

Tabela 5.8 - Comparação de estratégias de maquinagem CNC/RI

		1	2	3	4	5	6	7	8
									
Nº de linhas	CNC	46	94	58	194	2221	1687	85	80
	RI	42	91	54	163	2214	963	59	58
Tempo de execução [s]	CNC	169	179	173	177	238	588	190	242
	RI	169	172	169	176	237	595	183	239
Distância de maquinagem [mm]		3280	3506	3400	3555	4578	11725	3460	4856

Extensão dos programas e tempo de execução

Analisando os valores que constam na tabela anterior, conclui-se que:

- A trajetória em *Zig-zag*, possui o menor tempo de execução, menor extensão e menor distância de maquinagem.
- A extensão dos programas do robô foi, globalmente, inferior. Esta diferença é mais elevada para a estratégia de maquinagem alta velocidade e espiral real. Isto pode justificar-se pelo facto de envolveram movimentos circulares em maior número.
- Em ambos os sistemas a maquinagem mais extensa é a espiral de *Morph*, que consiste na utilização de múltiplos movimentos curtos lineares ao longo do percurso, daí a geração de um maior número de linhas. Note-se que a influência da grande extensão de linhas no tempo de maquinagem é quase nula devido à parametrização (G05 e zI) que atenua as variações de velocidade na passagem destes pontos;

- Visto os tempos serem semelhantes e as distâncias iguais, a velocidade média de maquinagem é aproximadamente igual, comparando o CNC e RI.

Percursos obtidos

Para os casos de estratégias de maquinagem que geram múltiplos segmentos lineares, o CNC evidencia perturbações nos percursos nas imediações das mudanças de direção, que é notório, quando as direções dos percursos estão alinhadas com os eixos de movimento (XY). Já no robô como envolve apenas juntas rotativas esse efeito não é tão visível. Na Figura 5.43 estão representados os percursos obtidos para as estratégias de maquinagem espiral com sobreposição constante e espiral com cantos retocados, para o CNC e RI.

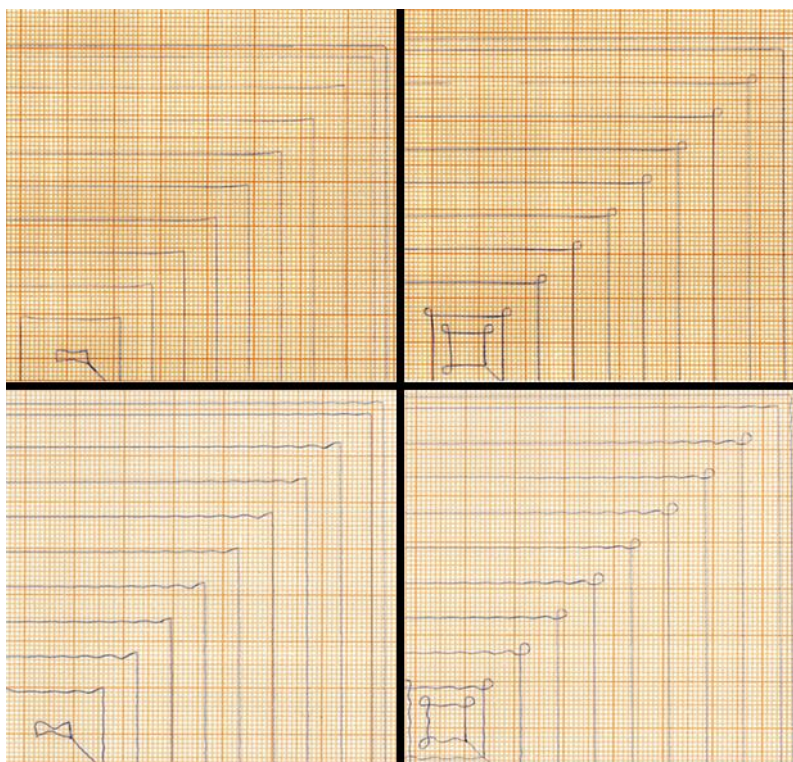


Figura 5.43 - Trajetórias obtidas no robô (em cima) e CNC (em baixo).

Síntese de resultados: estratégias de maquinagem

Tendo sido usada a função G05 no caso dos programas CNC e o parâmetro zI no caso do robô, a uma velocidade baixa verifica-se que o comportamento é similar em termos de tempo e proximidade dos percursos obtidos relativamente aos programados.

6 Conclusões e perspectivas de trabalhos futuros

O objetivo primordial desta dissertação passa, principalmente, por avaliar a influência das diferentes parametrizações aplicadas a um robô e a um CNC, em primeira instância para movimentos simples, lineares e circulares, seguida de uma extrapolação para casos mais complexos, como estratégias de maquinagem. Após a realização de vários ensaios e fazendo variar diversos parâmetros simultaneamente, ou alternando diferentes variáveis, foi possível retirar algumas ilações de todo o trabalho desenvolvido:

- No robô, o *zone* é um parâmetro decisivo na descrição da trajetória afetando a velocidade e o percurso executado pelo robô, assim, quanto maior for o seu valor, maior será o desvio em relação ao percurso programado e menor será a redução de velocidade relativamente à velocidade especificada;
- No robô, o parâmetro *AccSet* é importante na medida que limita a aceleração envolvida, melhorando a execução do traçado dos percursos;
- No CNC, a função arredondamento de aresta (G05), revela-se muito útil na manutenção da velocidade quando existem mudanças de direção, no entanto, o percurso é alterado para velocidades de avanço elevadas;
- No CNC, a função *look-ahead* é vantajosa na execução de trajetórias, na medida que, é possível especificar um erro de contorno, no entanto, também se verificou que esta é dependente da velocidade de avanço.

Em relação às funcionalidades disponíveis nos dois sistemas foi possível identificar funcionalidades equivalentes para evitar a paragem (redução a zero da velocidade) nos percursos, entre blocos sucessivos. No entanto o CNC disponibiliza um número mais elevado de funções para condicionar ou definir percursos de maquinagem (*e. g.* definição do círculo). Nas funções analisadas, o CNC evidenciou maior dependência da velocidade programada.

No que concerne às estratégias de maquinagem de caixas disponíveis e parametrizadas no Mastercam, a análise concentrou-se na extensão dos programas gerados para cada operação, nos seus tempos de simulação e, por último, nas trajetórias obtidas. Neste, verifica-se que:

- A execução dos percursos relativos às estratégias implementadas evidenciou os desvios associados aos movimentos em canto alinhados com os eixos de movimento da mesa;
- A utilização da função G05 e parâmetro *zI* permitiu obter resultados equivalentes nos tempos de execução a baixa velocidade com o CNC e com o robô;
- As diferenças na extensão dos programas de maquinagem obtidos, através do *software* Mastercam, devem-se a características próprias dos pós-processadores disponíveis.

Em termos de trabalhos futuros propõe-se:

- Em relação aos testes com estratégias de maquinagem poderiam ser feitos considerando a utilização da função *look-ahead* (G51), de modo a avaliar melhor as suas implicações em programas com um maior número de blocos;
- Fazer uma atualização do *software* do CNC de modo a permitir registar de forma automática a análise obtida com a função *Oscilloscope*; desse modo seria possível analisar, a influência das funções e parâmetros estudados nos perfis de velocidade em programas de maquinagem mais extensos (ex. estratégias de maquinagem);
- Relativamente aos testes efetuados poderiam ainda ser realizados em diferentes orientações e localizações no plano de trabalho;
- Recorrer a uma estrutura mais rígida no CNC, de forma a eliminar grande parte da perturbação, originada pela oscilação da mesma;
- Executar as estratégias de maquinagem com materiais diversificados com vista a um melhor conhecimento dos dois sistemas, de modo a incluir forças de corte.

Referências

1. J. Kending, P.V. *Machining*. 2002 2002 [cited 2018 20/02]; Available from: <http://www.mfg.mtu.edu/cyberman/>.
2. Suh, S.-H., et al., *Theory and design of CNC systems*. 2008: Springer Science & Business Media.
3. Pabla, B. and M. Adithan, *CNC machines*. 1994: New Age International.
4. FANUC. *CNC Systems*. 2017 [cited 2018 24/02]; Available from: <http://www.fanuc.eu/ro/en/cnc>.
5. FANUC. *Série de controlo CNC*. [cited 2018 24/02]; Available from: <http://www.fanuc.eu/pt/pt/cnc/controlos/série-controlo-cnc>.
6. FANUC. *Series 0i-MODEL F*. 2014; Available from: [http://www.fanuc.co.jp/en/product/catalog/pdf/cnc/FS0i-F\(E\)-02.pdf](http://www.fanuc.co.jp/en/product/catalog/pdf/cnc/FS0i-F(E)-02.pdf).
7. Anderton, J. *VIDEO: What You Need to Know About SINUMERIK CNC Controls*. 2016; Available from: <https://www.engineering.com/AdvancedManufacturing/ArticleID/13325/VIDEO-What-You-Need-to-Know-About-SINUMERIK-CNC-Controls.aspx>.
8. AG, S. *CNC controls*. [cited 2018 23-04]; Available from: <https://mall.industry.siemens.com/mall/en/WW/Catalog/Products/7500002?tree=CatalogTree>.
9. Siemens Industry, I. *SINUMERIK 840D sl*. 2014; Available from: <https://www.industry.usa.siemens.com/drives/us/en/cnc/systems-and-products/Documents/Brochure-SINUMERIK-840D-sl.pdf>.
10. CNC Software, I. *Mastercam*. 2018; Available from: <https://www.mastercam.com/en-us>.
11. *CIMdata Ranks Mastercam 1st in Installed Seats*. 2017; Available from: <https://www.mastercam.com/en-us/News/Press-Releases/CIMdata17>.
12. Mastercam. *Mastercam: Solutions*. 2018; Available from: <https://www.mastercam.com/en-us/Solutions>.
13. IFR. *Industrial Robots - Definition and Types*. 2016; Available from: https://ifr.org/img/office/Industrial_Robots_2016_Chapter_1_2.pdf.
14. Aryania, A., et al. *New trends in industrial robot controller user interfaces*. in *Cognitive Infocommunications (CogInfoCom), 2012 IEEE 3rd International Conference on*. 2012. IEEE.
15. *Executive Summary World Robotics 2017 Industrial Robots*. 2017; Available from: https://ifr.org/downloads/press/Executive_Summary_WR_2017_Industrial_Robots.pdf.
16. Iglesias, I., M. Sebastián, and J. Ares, *Overview of the state of robotic machining: Current situation and future potential*. *Procedia engineering*, 2015. **132**: p. 911-917.

17. Matsuoka, S.-i., et al., *High-speed end milling of an articulated robot and its characteristics*. Journal of materials processing technology, 1999. **95**(1-3): p. 83-89.
18. Klimchik, A., et al., *Efficiency evaluation of robots in machining applications using industrial performance measure*. Robotics and Computer-Integrated Manufacturing, 2017. **48**: p. 12-29.
19. Guo, Y., H. Dong, and Y. Ke, *Stiffness-oriented posture optimization in robotic machining applications*. Robotics and Computer-Integrated Manufacturing, 2015. **35**: p. 69-76.
20. Schneider, U., et al. *Experimental Investigation of Sources of Error in Robot Machining*. 2013. Berlin, Heidelberg: Springer Berlin Heidelberg.
21. Stäubli. *High speed machining (hsm) robots*. 2018 [cited 2018 12/03]; Available from: <https://www.staubli.com/pt-pt/robotics/product-range/6-axis-scara-picker-industrial-robots/application-pack/high-speed-machining-robots/>.
22. Robotics, A. *IRB 6660*. 2018 [cited 2018 22/04]; Available from: <http://new.abb.com/products/robotics/industrial-robots/irb-6660>.
23. Motoman. *MH80 II*. 2017 [cited 2018 12/03]; Available from: <https://cdn2.hubspot.net/hubfs/366775/Robots/MH80II.pdf>.
24. KUKA. *KR 600 FORTEC*. 2018; Available from: <https://www.kuka.com/en-de/products/robot-systems/industrial-robots/kr-600-fortec>.
25. Fundilusa. *CNC Robot Milling*. 2017 [cited 2018 21/04]; Available from: <https://vimeo.com/215432302>.
26. R, A.V. [cited 2018 08 03]; Available from: <https://avr-global.com/en/>.
27. Robotmaster. *Robot programming software addresses complex material removal application*. 2017; Available from: <http://www.robotmaster.com/en/success-stories/avr-vision-robotics>.
28. R, A. *Discover Robotic Blade Polishing - AV&R Aerospace*. 2018; Available from: https://www.youtube.com/watch?time_continue=36&v=cjIL2GyUBJw.
29. JWQ. Available from: <https://www.johnsons-wellfield.co.uk>.
30. *Tudor Square*. 2008-2010; Available from: <https://broadbent.studio/tudor-square-2/>.
31. Pan, Z., et al., *Recent progress on programming methods for industrial robots*. Robotics and Computer-Integrated Manufacturing, 2012. **28**(2): p. 87-94.
32. *Programming Robots*. Available from: <http://thnet.co.uk/thnet/robots/33.htm>.
33. Robotmaster. *Applications*. 2017 [cited 2018 15/03]; Available from: <http://www.robotmaster.com/en/applications>.
34. Kuka Quantec *CNC Robot*. 2014 [cited 2018 15/03]; Available from: https://www.youtube.com/watch?v=Kc2dP_Wh6ow.
35. Inc., H. *Robotmaster at IMTS 2014*. 2014 [cited 2018 15/03]; Available from: https://www.robotics.org/content-detail.cfm/Industrial-Robotics-News/Robotmaster-at-IMTS-2014/content_id/4999.
36. Automation, F., *Driver DDS: Software manual*. 2015(Versão 1502).
37. Gonçalves, D.J.d.R., *Montagem e Teste de um CNC Modular*, in *Automação*. 2015, FEUP.

38. Usop, Z., et al., *Measuring of positioning, circularity and static errors of a CNC Vertical Machining Centre for validating the machining accuracy*. Measurement, 2015. **61**: p. 39-50.
39. Automation, F., *CNC 8055 - M &EN - Programming Manual*. 2014: p. 34-36.
40. Automation, F., *CNC 8055M: Manual de Programação*. (Versão 0004).
41. ABB, *Technical reference manual: RAPID instructions, functions and data types*. ABB Robotics, Västerås, Sweden, 2014.
42. OSG. *CONDIÇÕES RECOMENDADAS DE USINAGEM PARA FRESAS HSS-Co*. Available from: <http://www.osg.com.br/v4/downloads/vc/P450.pdf>.

ANEXO A: Funções de programação do CNC

Função	M	D	V	Significado	Seção
G00	*	?	*	Posicionamento rápido	6.1
G01	*	?	*	Interpolação linear	6.2
G02	*		*	Interpolação circular (helicoidal) direita	6.3
G03	*		*	Interpolação circular (helicoidal) esquerda	6.3
G04				Temporização/Detenção da preparação de blocos	7.1, 7.2
G05	*	?	*	Arredondamento de aresta	7.3.1
G06			*	Centro circunferência em coordenadas absolutas	6.4
G07	*	?		Aresta viva	7.3.2
G08			*	Circunferência tangente à trajetória anterior	6.5
G09			*	Arco definido por 3 pontos	6.6
G10	*	*		Cancelamento de espelhamento	7.5
G11	*		*	Espelhamento em X	7.5
G12	*		*	Espelhamento em Y	7.5
G13	*		*	Espelhamento em Z	7.5
G14	*		*	Espelhamento nas direções programadas	7.5
G15	*		*	Seleção do eixo longitudinal	8.2
G16	*		*	Seleção plano principal por duas direções	3.2
G17	*	?	*	Plano principal X-Y	3.2
G18	*	?	*	Plano principal Z-X	3.2
G19	*		*	Plano principal Y-Z	3.2
G20				Definição limites inferiores zonas de trabalho	3.7.1
G21				Definição limites superiores zonas de trabalho	3.7.1
G22		*		Habilitação/desabilitação zonas de trabalho	3.7.2
G23		*		Ativação do copiado	16.3
G24		*		Ativação do digitalizado	16.6
G25		*		Desativação do copiado/digitalizado	16.5
G26		*		Calibração da sonda do copiado	16.2
G27		*		Definição do contorno de copiado	16.4
G28	*		*	Seleciona a Segunda árvore	5.4
G29	*	*		Seleciona a Árvore principal	5.4
G28-29			*	Comutação de eixos	7.9
G30	*		*	Sincronização de eixos-árvore (defasagem)	5.5
G32	*		*	Avanço F como função inversa do tempo	6.14
G33	*		*	Rosqueamento eletrônico	6.12
G36		*	*	Arredondamento de arestas	6.10
G37		*	*	Entrada tangencial	6.8
G38		*	*	Saída tangencial	6.9
G39		*	*	Introdução automática de chanfros	6.11
G40	*	*		Cancelamento de compensação radial	8.1
G41	*		*	Compensação radial ferramenta à esquerda	8.1
G42	*		*	Compensação radial ferramenta à direita	8.1
G43	*	?	*	Compensação longitudinal	8.2
G44	*	?	*	Cancelamento da compensação longitudinal	8.2
G45	*		*	Controle tangencial (G45)	6.15
G47	*	*	*	Deslocar ferramenta conforme sistema coord. da ferramenta ...	17.2
G48	*		*	Transformação TCP	17.3
G49	*		*	Definição do plano inclinado	17.1
G50	*	*	*	Arredondamento de aresta controlado	7.3.3
G51	*		*	Look-Ahead	7.4
G52		*	*	Movimento contra ponto	6.13
G53		*	*	Programação referente ao zero máquina	4.3
G54	*	*	*	Deslocamento de origem absoluta 1	4.4.2
G55	*	*	*	Deslocamento de origem absoluta 2	4.4.2
G56	*	*	*	Deslocamento de origem absoluta 3	4.4.2
G57	*	*	*	Deslocamento de origem absoluta 4	4.4.2
G58	*	*	*	Deslocamento de origem aditivo 1	4.4.2
G59	*	*	*	Deslocamento de origem aditivo 2	4.4.2
G60		*	*	Usinagem multiplíce em linha reta	10.1
G61		*	*	Usinagem multiplíce formando um paralelogramo	10.2
G62		*	*	Usinagem multiplíce em malha	10.3
G63		*	*	Usinagem multiplíce formando uma circunferência	10.4
G64		*	*	Usinagem multiplíce formando um arco	10.5
G65		*	*	Usinagem programada mediante uma corda de arco	10.6

Função	M	D	V	Significado	Seção
G66			*	Ciclo fixo de cavidades com ilhas	11.1
G67			*	Operação de desbaste de cavidades com ilhas	11.1.2
G68			*	Operação de acabamento de cavidades com ilhas	11.1.3
G69	*		*	Ciclo fixo furação profunda com passo variável	9.5.1
G70	*	?	*	Programação em polegadas	3.3
G71	*	?	*	Programação em milímetros	3.3
G72	*		*	Fator de escala geral e particulares	7.6
G73	*		*	Rotação de coordenadas	7.7
G74			*	Buscar marca de referência	4.2
G75			*	Movimento com apalpador até tocar	12.1
G76			*	Movimento com apalpador até deixar de tocar	12.2
G77	*		*	Acoplamento eletrônico de eixos	7.8.1
G77S	*		*	Sincronização de eixos-árvore	5.5
G78	*	*	*	Cancelamento do acoplamento eletrônico	7.8.2
G78S	*	*	*	Anulação da sincronização de eixos-árvore	5.5
G79			*	Modificação de parâmetros de um ciclo fixo	9.2.1
G80	*	*	*	Cancelamento de ciclo fixo	9.3
G81	*		*	Ciclo fixo de furação	9.5.2
G82	*		*	Ciclo fixo de furação com tempo de espera	9.5.3
G83	*		*	Ciclo fixo furação profunda com passo constante	9.5.4
G84	*		*	Ciclo fixo de rosqueamento com macho	9.5.5
G85	*		*	Ciclo fixo de alargamento	9.5.6
G86	*		*	Ciclo fixo mandrilamento com retrocesso em G00	9.5.7
G87	*		*	Ciclo fixo de cavidade retangular	9.5.8
G88	*		*	Ciclo fixo de cavidade circular	9.5.9
G89	*		*	Ciclo fixo mandrilamento com retrocesso em G01	9.5.10
G90	*	?	*	Programação em absoluto	3.4
G91	*	?	*	Programação em incremental	3.4
G92				Pré-seleção de cotas / Limitação velocidade da árvore principal	4.4.1
G93				Pré-seleção da origem polar	4.5
G94	*	?	*	Avanço em milímetros (polegadas) por minuto	5.2.1
G95	*	?	*	Avanço em milímetros (polegadas) por rotação	5.2.2
G96	*		*	Velocidade do ponto de corte constante	5.2.3
G97	*		*	Velocidade de centro da ferramenta constante	5.2.4
G98	*		*	Volta plano de partida ao final ciclo fixo	9.5
G99	*		*	Volta plano de referência ao final ciclo fixo	9.5